

IMPLEMENTATION AND TESTING
OF THE NPS DATA COMMUNICATION RING

John S. Meserve

DODLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

IMPLEMENTATION AND TESTING
OF THE NPS DATA COMMUNICATION RING

by

JOHN S. MESERVE

and

ROBERT STEWART

JUNE 1975

THESIS ADVISOR:

GARY M. RAETZ

Approved for public release; distribution unlimited.

T167972

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Implementation and Testing of the NPS Data Communication Ring		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June, 1975
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) John S. Meserve and Robert Stewart		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June, 1975
		13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (If different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The Naval Postgraduate School Ring Communication Network is a project which, when fully operational, will allow connection of designated computer facilities at the school. The capability provided to these facilities will be the transmission of data or messages to one another, which will allow		

Implementation and Testing
Of the NPS Data Communication Ring

by

John S. Meserve
Lieutenant Commander, United States Navy
B.S., Merchant Marine Academy, 1963
and

Robert E. Stewart
Lieutenant, United States Navy
B.S., Purdue University, 1969

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 1975

ABSTRACT

The Naval Postgraduate School Ring Communication Network is a project which, when fully operational, will allow connection of designated computer facilities at the school. The capability provided to these facilities will be the transmission of data or messages to one another, which will allow sharing of resources. The main considerations in design of the network were modularity, to simplify construction and testing, and common hardware interfaces between the ring and each computer facility. To provide the capability to test all hardware/software functions test procedures were developed, and a hardware test module was constructed.

The basic design of the ring network and test module is described. Test procedures and results are presented, as well as a proposal for a system modification, and recommendations concerning follow-on development of the system.

TABLE OF CONTENTS

I.	INTRODUCTION-----	8
	A. BACKGROUND-----	8
	B. TERMINOLOGY-----	8
	C. DESIGN CONCEPTS-----	9
II.	FUNCTIONAL COMPONENTS OF A NODE-----	12
	A. REPEATER-----	12
	B. RING INTERFACE-----	12
	C. HOST ADAPTOR-----	15
	D. PARALLEL DATA ADAPTOR-----	16
	E. MICRO-CONTROLLER-----	16
III.	MESSAGE HANDLING-----	17
	A. MESSAGE FORMATS-----	17
	B. TOKENS-----	17
	C. RI CONTROL-----	20
IV.	TIMING-----	23
V.	TEST PROCEDURES-----	25
	A. INITIAL CONSIDERATIONS-----	25
	B. TEST MODULE-----	26
VI.	CONCLUSIONS AND RECOMMENDATIONS-----	35
	A. STATE OF THE PROJECT-----	35
	B. RECOMMENDATIONS-----	37
	APPENDIX A LIST OF COMMON ABBREVIATIONS-----	41
	APPENDIX B RI PROCEDURAL FLOWCHARTS-----	48
	APPENDIX C HA PROCEDURAL FLOWCHARTS-----	63
	BIBLIOGRAPHY-----	82
	INITIAL DISTRIBUTION LIST-----	83

LIST OF FIGURES

1.	EXAMPLE NPS DATA COMMUNICATION RING CONFIGURATION-----	10
2.	NODE CONFIGURATION AND INTERCONNECTIONS--	13
3.	MESSAGE FORMATS -----	18
4.	TYPICAL RING MESSAGE-----	19
5.	LOCAL COMMAND MESSAGE FORMATS-----	22
6.	TEST MODULE CONFIGURATION-----	27
7.	STATUS PANEL WIRING DIAGRAM-----	28
8.	SIMULATION PANEL WIRING DIAGRAM-----	29
9.	STATUS PANEL, SIMULATION PANEL, AND BUFFER BOARD INTERCONNECTING WIRING DIAGRAM----	30
10.	HA, RI, AND BUFFER BOARD INTERCONNECTING WIRING DIAGRAM-----	31
11.	BUFFER BOARD PHYSICAL LAYOUT-----	32
12.	RESET LINE MODIFICATION TO RI MICRO-CONTROLLER-----	38

ACKNOWLEDGEMENT PAGE

DEDICATED TO MARY AND KATHY
WHO HAVE PROVED THAT
PATIENCE IS WASTED ON THE YOUNG

I. INTRODUCTION

A. BACKGROUND

The Naval Postgraduate School Data Communication Ring has been under design and construction for over two years. Originally conceived by Lieutenant Keith Hirt, and described in his master's thesis, "A Prototype Ring-Structured Computer Network Using Micro-Computers," [Ref. 4], the theory of such a network was formalized. The hardware implementation was then undertaken in separate projects. The ring interface and its associated micro-controller was implemented by Ensign Michael J. Harris, and described in his master's thesis, "A Prototype Ring Interface for the NPS Data Communication Ring." [Ref. 3]; while concurrently in time the Ring Interface/Host Adaptor and its associated micro-controller was implemented by Lieutenant Commander Eberhard O. Wortmann, and described in his master's thesis, "Design and Implementation of a Ring Interface/Host Adaptor for an IBM System 360." [Ref. 8]. Details of the theory and implementation previously accomplished will not be reproduced in this report; rather, those items required for clarity will be paraphrased or summarized where necessary.

B. TERMINOLOGY

Much specialized terminology has been introduced in the various papers that deal with the ring network. This creates some difficulty in understanding the protocols that exist. Another area of confusion exists concerning the control lines that connect the Ring Interface (RI) to the Host Adaptor (HA). Although their functions are well

defined, labeling differs between the Harris and Wortmann papers. In order to have a ready reference of common terms and abbreviations that apply to this system, Appendix A is included in this report. This Appendix lists all commonly used abbreviations and terms in alphabetical order, and gives a brief definition or explanation. In addition, where considered necessary, a cross reference is made to any other abbreviation or term which is applied to the same functional entity.

C. DESIGN CONCEPTS

A Data Communication Ring consists of a Data Transmission Line (DTL), Ring Interfaces (RI), and Host Adaptor (HA) if required. Each Host Processor and Ring Interface constitute a Node in the network. Figure 1 shows a possible configuration of such a network at the Naval Postgraduate School. The Data Transmission Line transmits data, in the form of messages, serially and unidirectionally around the ring. The Nodes have the ability to connect and disconnect from the ring without disturbing the flow of data. The Host Processors are the computers, terminals, disk systems, etc., to and from which data is transmitted. In essence then, the ring network allows the Host Processors to communicate data to one another and therefore allows sharing of resources in the ring.

Key requirements in the design of the NPS ring were reliability and economy. To maintain high reliability no Node is given ultimate control of the ring, therefore, failure of any Node will not cause failure of the network. A control hierarchy is built into the system via a timer at each RI. A RI will take control of the ring automatically if, and only if, there is no other RI, higher in the hierarchical

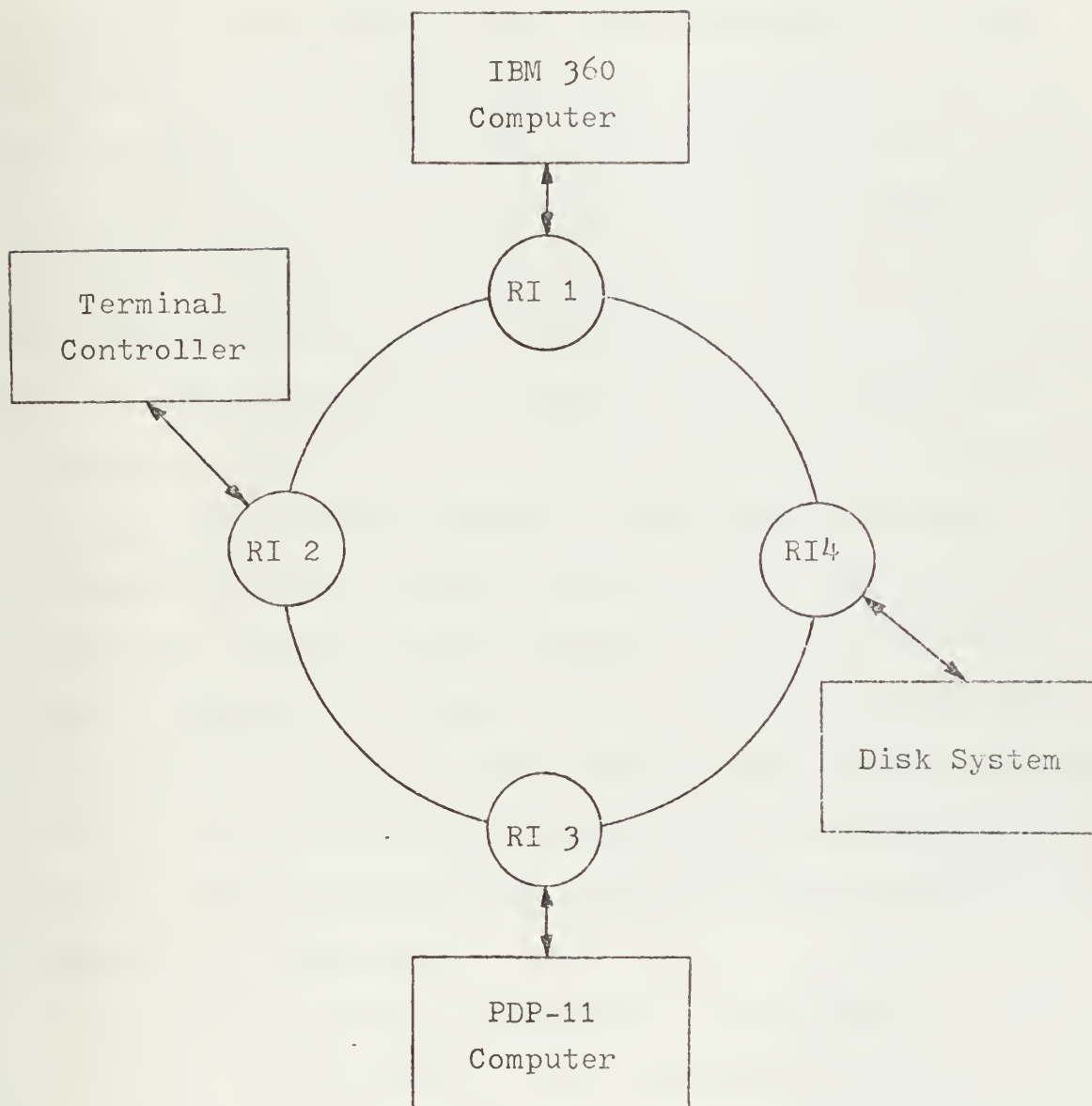


FIGURE 1. Example NPS Data Communication Ring Configuration.

order, to take control. The important facet is that only one RI is in control at any time, eliminating multiple message flow or control on the ring. Control of the ring is designed to pass cyclically around the ring from Node to Node. When a Node has control of the ring any host processor residing at that Node is then allowed to transmit data to any other host on the ring. If any process in the host does not wish to transmit a message then control of the ring is passed to the next Node on the ring. If a message arrives for a host processor the RI signals his host processor to prepare to receive the data. The data is buffered in the HA to eliminate problems of differing data transmission rates. The RI signals the end of message to the host processor and continues to monitor the ring either for another message addressed to its host, or for a signal that it can take control of the ring. Messages are not removed from the ring but merely copied. Therefore, more than one host processor can be addressed by a message. The sending Node, however, does remove the message from the ring, simultaneously checking status bits to determine if the message was received error free. Message formatting will be covered in detail later in this report.

Cost constraints were met by using micro-controllers to control all functions of the RI and HA. This reduced hardware requirements to a minimum. As part of this project a second RI and HA with their associated micro-controllers were locally constructed. The cost of the integrated circuit components was less than \$500.00 (not including the cost of four programmable read-only memories (PROM) which were on hand).

Purchase of materials would be considerably less if larger quantities were ordered; and when the system is implemented, the PROM in each micro-controller would be replaced by a metal masked ROM at a considerable cost reduction.

II. FUNCTIONAL COMPONENTS OF A NODE

Each node can be logically subdivided into functional hardware components which have distinct responsibilities. Figure 2 shows such a subdivision with the associated interconnections. Following is an overview of the functional responsibilities of the components.

A. REPEATER

The Repeater provides the necessary signal boost to drive the messages over long cable lengths. It is designed to be directly connected to the ring, receive the signals, recover clocking information, and pass on (cleaned up, reshaped) data to the outbound cable. The design of the Repeater is dependent on several characteristics of the actual hardware. That is; the cable type, cable length between Nodes, type of drivers/receivers, and ring speed all affect Repeater design. The actual hardware design has been deferred until more is known about these characteristics.

B. RING INTERFACE

The heart of the ring network is the Ring Interface, which is Host Processor independent from a design standpoint. The RI has three major responsibilities:

1. Accept and comply with all control signals sent by the host processor. This includes connecting and disconnecting from the ring.

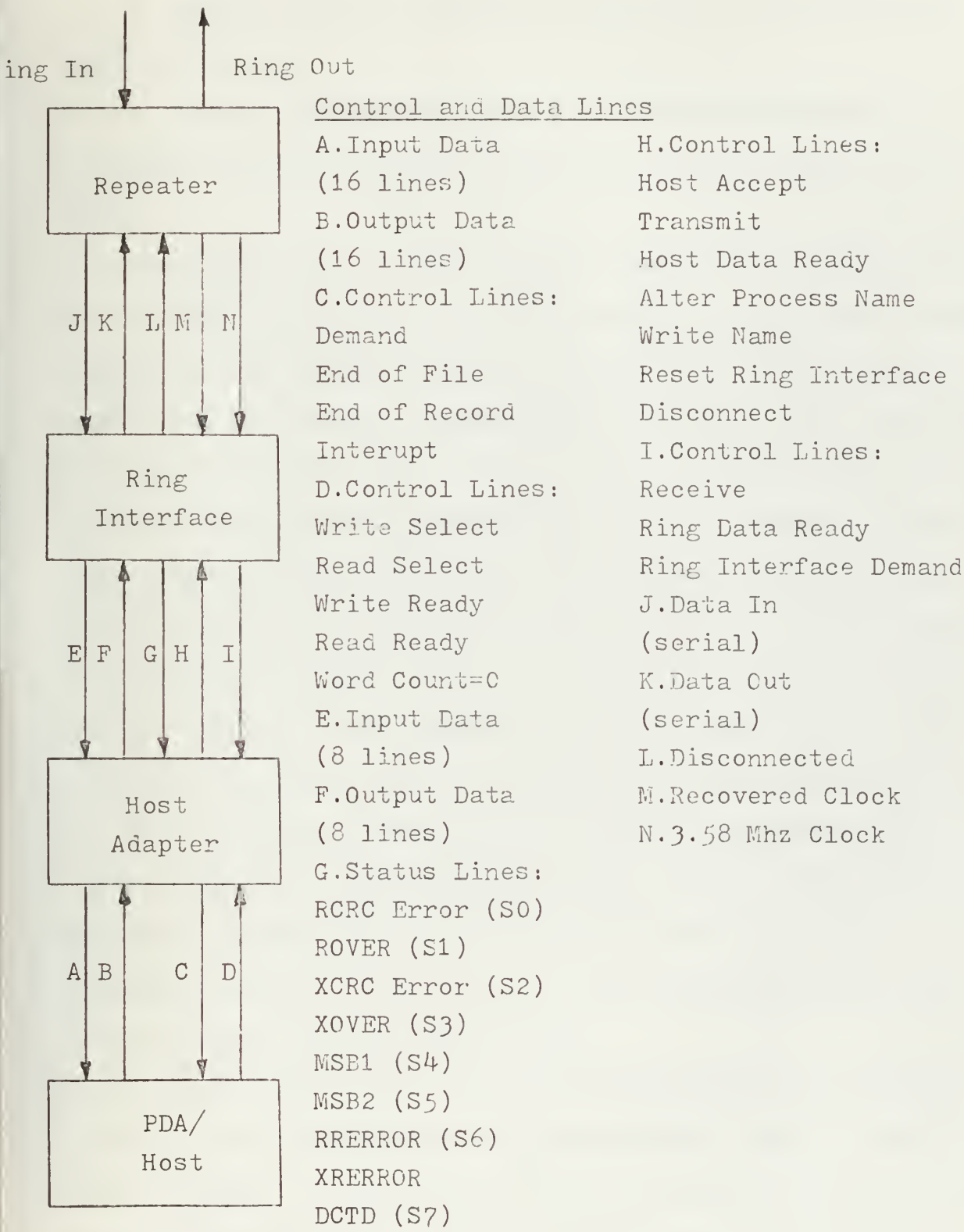


FIGURE 2. Node Configuration and Interconnections

2. Accept, deliver, retransmit, and evaluate data and tokens to and from the ring.

3. Continuously check for errors in outgoing and incoming messages, keeping the host processor informed by the use of status lines.

In the transmit mode the RI receives data in 8 bit parallel format. The message is sent to the ring in the correct order, with data encoded (coding formats are discussed in detail later in this report) and converted to serial format for transmission to the ring. While transmitting this data the RI continuously checks for errors. A bipolar violation error is a series of three like bits (lll) or more in sequence. These sequences are not allowed in data bytes, and are an indication to the RI that either an error has been detected or that one of the tokens has been detected. The RI is able to determine the difference between an error and a token. At the same time the outgoing serial data stream is being continuously divided by a polynomial technique handled by the RI. At the end of the data stream the remainder is transmitted as a Cyclic Redundancy Check Character (CRCC). At the receiving Node this same calculation is performed on the incoming data bytes, including the CRCC. If there were no errors introduced into the data stream the new remainder will be zero.

In the receive mode the RI receives serial data from the Repeater, decodes it, and converts it to 8 bit parallel format, which is passed on to the HA buffer. A handshaking routine is carried on between the RI and HA to insure proper handling of the data bytes. The RI has the responsibility to determine what messages are addressed to one or more processes resident in the Node. In the receive mode the RI

again maintains a constant check for bipolar violation errors and the CRC check, keeping the host advised by status lines. At the end of a message the RI updates the message status bits which are included in the retransmitted data stream that goes back to the originating Node. The message status bits are discussed in detail later in this report.

C. HOST ADAPTOR

The Host Adaptor (HA) acts as an interpreter between a host processor and the RI. The HA design for a particular Node therefore is dependent upon the host being served. The HA implemented by Wortmann was designed to interface with the IBM 360 hardware, specifically the IBM 360 Parallel Data Adaptor. In some cases, especially mini-computer applications, the host processors could be directly linked to the Ring Interface, with the HA functions being accomplished by host software. The original concepts for the NPS ring by Hirt [Ref. 4] envisioned common RI's with mini-computers acting as the adaptors to the host processors, but this concept has been modified. Unless specifically stated otherwise, all discussion in this paper of the HA design or functional responsibilities will refer to the IBM 360 HA.

The HA converts 16 bit parallel data from the PDA to 8 bit parallel data for the RI. This sequence is reversed in the receive mode. A major function is buffering of the data in a first-in-first-out (FIFO) buffer, which consists of a 1024 x 8 bit random access memory. Through the use of hardware this RAM is configured to allow data to "fall through" in FIFO order in both directions, depending on whether the Node is in a receive or transmit mode. The major reason for

this buffer is to allow for differing rates of data flow in the various components of the system.

The HA also interprets and passes on messages (commands) from the host to the RI. These Local Command Messages (LCM) are used by the host to control the actions of the RI.

D. PARALLEL DATA ADAPTOR

The PDA is an IBM hardware device which acts as an interface between external devices, and the hardware of the IBM 360. Software interface with the ring is discussed in the Johnson/Kirkham Thesis [Ref. 5].

E. MICRO-CONTROLLER

To minimize hardware costs and allow for ease of prototype modification, control of the functions accomplished by both the RI and HA is maintained by micro-controllers (MCs). The MCs used in the prototype Node are very similar in hardware design, with different programs residing in the respective PROMs. Basically, the MCs are able to test logic levels of up to 32 input ports, set logic levels on any of 32 output ports, and receive or transmit over 8 parallel data lines. The instruction set is quite simple and consists of unconditional jump, conditional jump, jump external and output instructions. Particulars of hardware design are given by Brubaker in Ref. 1. Flowcharts of the micro-controller programs are listed in Appendix B and C.

III. MESSAGE HANDLING

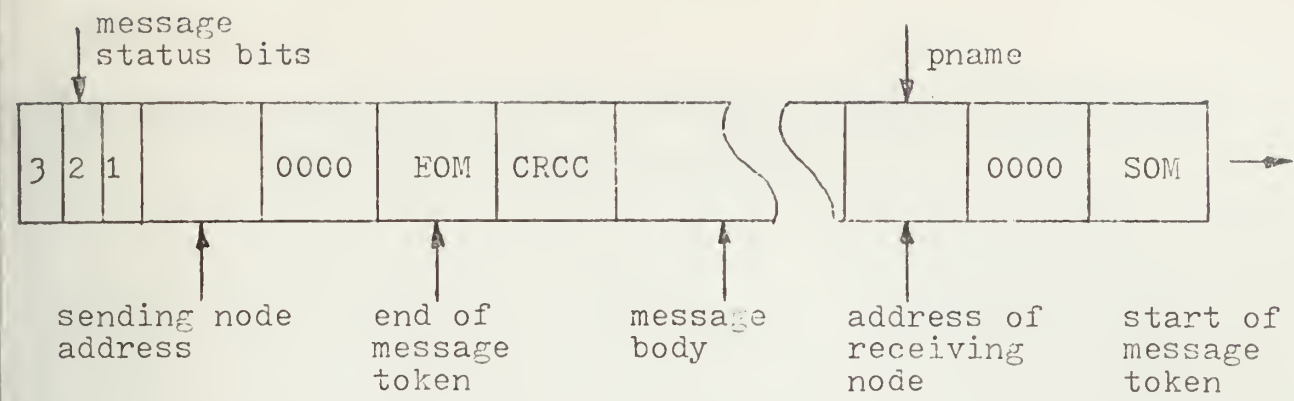
A. MESSAGE FORMATS

Message formats change considerably between the PDA, HA, RI functional units and the ring. Data is sent in 16 bit parallel format to, and from, the PDA. Between the HA and RI the data is transmitted in 8 bit parallel format. All handshaking techniques are handled by control and status lines between these units. The RI changes the format of data and encodes it for transmission on the ring. Figure 3 shows the format of messages on the ring, Message Status Bit definition, and coding format. Note that each data bit is encoded into two bits. For example, a data byte (10011001) would be encoded and would be transmitted as 16 serial bits (1001011010010110) on the ring. Note also the low order bits are transmitted first. Figure 4 shows an example message, excluding the data bits.

B. TOKENS

Control of the ring and message signals are accomplished through the use of tokens, which are listed in Figure 3. All tokens start with a bipolar violation (111---) which is interpreted by the RIs to mean that possibly a token follows and allows for decoding. The uses of the three tokens are now described:

1. CTL - The Control Token signals to an RI that it may take control of the ring and send a message if the host desires. If no message is ready for transmission, or when it is through transmitting a message, the RI places a CTL token on the ring, which passes to the next Node "downstream". Thus, the CTL token allows one Node to be in charge of the ring at a time.



TOKENS:

TOKEN	MEANING	BINARY REPRESENTATION
SOM	Start of Message	11101100
EOM	End of Message	11100101
CTL	Control	11101010

MESSAGE STATUS BITS:

BIT	MEANING IF ZERO (0)	MEANING IF ONE (1)
1	No RI matched and accepted message	At least one RI matched and accepted message
2	No RI matched without accepting	At least one RI matched but did not accept
3	No target RI recognized a CRC error	At least one RI recognized a CRC error

CODING FORMAT:

0 is encoded to 01
 1 is encoded to 10

FIGURE 3. Message Formats

SOM TOKEN: Alerts all nodes a message follows. Padded with zeros for clocking purposes.

PNAME: Address of host processor message is intended for.
In this example address is node 5 (decimal). Low order bits sent first after encoding.

DATA: Encoded message body.

CRC: CRC Character. The remainder following the polynomial division accomplished by the Ring Interface

EOM TOKEN: End of message token. Alerts node that the message has ended, and to send status bits.

ADDRESS: This coded byte is the address of the sending node.

MSB: Message status bits.

FIGURE 4. Typical Ring Message.

Each RI has a "time-out clock" which is used to start up the ring, prevent hogging of the ring by one Node and prevent multiple messages on the ring simultaneously, by generating CTL tokens at the appropriate time. Details of the time-out feature are discussed in the timing section of this report.

2. SOM - The Start of Message Token alerts all Nodes that a message follows. The SOM resets the time-out clocks as it is received. Because tokens are only 8 bits in length and the counters in the RIs are modulo sixteen, the tokens are padded with eight zeros.

3. EOM - The End of Message Token alerts the RI that the message body is over and that the Node address (sending) and status bits follows.

C. RI CONTROL

The RIs then, use the tokens to get, and pass, control of the ring; as a signal that a message follows; and as a message delimiter. Receive and transmit sequences will now be discussed.

1. Receive Sequence

When an RI recognizes a SOM Token and recognizes that the message is addressed to a resident process in its host it begins the receive sequence. While constantly checking for three identical transmission bits in a row (bipolar violation) the RI uses the reception clock (clocked once for each incoming bit) to time the bits. After 16 bits have been clocked, a serial-in-parallel-out shift register is triggered. The 16 bits are decoded (the first bit of each pair is the data bit) and passed to the HA. After the EOM Token is recognized, the RI sets the appropriate bits into the message status bits. As this

data is being read into the FIFO buffer, the HA simultaneously reads the data and transfers it to the host in 16 bit parallel format as rapidly as the host can accept the information.

2. Transmit Sequence

In the transmit sequence two types of messages can be sent from the host; the local command message (LCM), and the regular transmit message (TM). The LCM never reaches the ring, but rather is used by the host to control the RI. Figure 5 lists valid LCMs, with the code used by the HA to differentiate the messages, and the functions accomplished by each LCM. Each LCM is two bytes in length, whereas all TMs are over two bytes. The host adaptor uses this fact to start the decoding process. Any time the HA receives a message from the host of one word (two bytes) it assumes a LCM is being transmitted and decodes the appropriate four bits of the first byte of the word. The command is then passed to the RI for action.

As stated, an LCM never reaches the ring but is used for local control. The transmission message, however, causes the RI to shift into the transmit sequence. As soon as the RI gets control of the ring (via a CTL token) it places an SOM token on the ring, followed by an address byte and all data bytes. The data bytes are automatically encoded for transmission. During this transmit sequence the host adaptor alternately gets data from the PDA until the final two data bytes are sent, and outputs the bytes to the RI. Again, the buffer in the host adapter allows for any speed variance in shifting the data through the Node to the ring. Error checking of the message is an ongoing function of both the sending and receiving ring interfaces. After the message passes around the ring the sending Node removes

BIT CODE OF

1ST BYTE

<u>LCM</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>FUNCTION</u>
WRITENAME	0	0	0	0	Turns on appropriate bit in PNAME memory. See note.
CLEARNAME	1	0	0	0	Erases the appropriate bit in PNAME memory.
DISCONNECT	0	1	0	0	Signals the RI to disconnect from the ring network.
CONNECT	1	1	0	0	Signals the RI to connect to the ring.
RESET RI	1	0	1	0	Strobes reset line to RI
STATUS REQUEST	0	0	1	0	Requests status bytes from the RI in case of an error.

Note: The HA decodes bits 4 through 7 (other bits are ignored) of the first byte sent from the host to determine which LCM is being transmitted. In the case of WRITENAME or CLEARNAME the second byte is the address of the bit written into or erased from PNAME memory. After this interpretation the host adapter passes on the LCM with appropriate handshaking to the ring interface for action.

FIGURE 5. Local Command Message Formats.

it from the ring, finally checking the message status bits for correct reception. This information is passed on to the host for any action required. As an example, if a host transmitted a message and was notified by its RI that the receiving Node detected an error in the received bits, the originating host might retransmit the message the next time it had control of the ring.

IV. TIMING

Timing and data transmission rates on the ring and within the components of a Node have been discussed in the various references. Presented here is an overall discussion of expected ring data transmission rates and timing problems.

The rate at which data will eventually flow around the ring network is a function of how fast the hardware can send the data, characteristics of the data transmission line, and finally, how fast the data can be received at a target Node. In Reference 2, Brubaker discussed various data transmission line possibilities, and suggests that line speed will be limited to about 250,000 bits per second. This limitation is primarily based on the fact that any cable acts as a low pass filter. As the bits are placed on the line at a faster rate less definition can be detected in the high and low peaks, until finally "one" and "zero" bits can no longer be distinguished. This data rate translates to a 125,000 bps actual data rate due to encoding of all bits into two bits. Reception/transmission rate of Nodes is another limiting factor, the assumption being made that all host processors will be able to handle data flow at least as fast as the envisioned ring. Within a Node then, the critical component is considered to be the RI, specifically the RI micro-controller.

Program design requires that the micro-controller execute several instructions between the reception or transmission of data bits. This allows the MC to test and set flags in the interval.

Brubaker [Ref. 1], considers the maximum MC cycle rate to be about 1.1 usec. To allow for four cycles from the MC between each bit, the ring interface is then limited to a cycle rate of about 4.5 usec. This translates to a rate of about 220,000 baud, or about 110,000 information bits per second.

This is well within the range allowed by the data transmission line. To maintain these cycle rates, a crystal clock is proposed to reside in the repeater. This clock would provide transmit clock pulses to the RI micro-controller, and after proper division would also provide transmit clock pulses to the RI. Although the host adaptor and its micro-controller run independently from the RI, cycle times which varied greatly from the MC could cause underflow or overflow of the FIFO buffer. This would cause an error condition whenever the RI is forced to wait for the IIA. Therefore, it is recommended that the HAMC be driven by the same clocking pulse that drives the RIMC.

Clocking during message reception is accomplished by a recovery clock, which also resides in the repeater. This "clock" pulse is generated and synchronized by the incoming data bits. During a receive sequence the RI must immediately start sending the data bytes to the IIA FIFO buffer. If this buffer overflows an error condition occurs as previously noted. If data is received at the maximum designed transmission rate, then the buffer is being filled at a rate of 13,750 bytes per second. This allows the resident host over 75 ms to initiate its receive sequence and start taking the data out of the buffer.

The timeout feature of each Node is discussed in detail by Harris on pages 20-24 of Ref. 3. The circuitry to handle this time-out clock has been designed and tested, but is not presently in operation in the prototype Node. The time-out clock prevents any Node from tying up the ring for long periods of time, and allows for starting and restarting of the ring. Each time-out clock has a unique delay time to prevent simultaneous control of the ring. This delay is varied by varying a capacitor and resistor used in conjunction with a timer. Actual implementation of this timer has been deferred until more is known about number of Nodes planned for the ring and proposed message lengths.

V. TEST PROCEDURES

At the inception of this thesis the status of the NPS ring system was as follows:

1. Basic design of hardware/firmware elements of a Node was complete.
2. Hardware implementation of the RI and HA and their associated micro-controllers was complete with some initial testing accomplished.

The next logical step then, was to thoroughly test each unit singly, followed by integration of the units and testing the resulting Node for inter-module protocol handling.

A. INITIAL CONSIDERATIONS

To prepare for future testing, construction of a second Node was conducted. This accomplished three purposes:

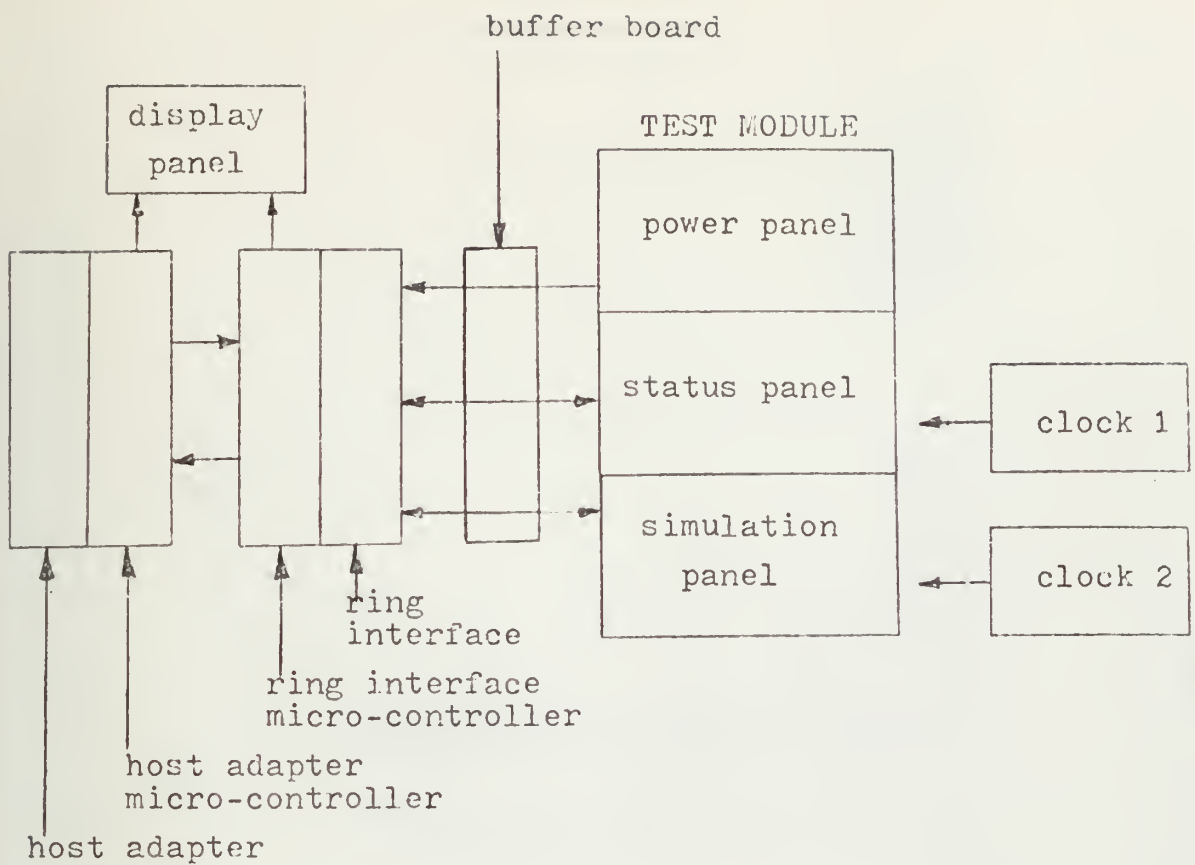
1. As a learning tool the wiring practice helped with familiarization of the components, and brought out some less than clear details presented in reference material.

2. The second host adaptor was constructed because it was felt that very minor hardware modifications would be required to modify it for use with other host processors.

3. Any real application would require at least two Nodes. Initial work included verifying the PROM programs by checking against the original program paper tapes. Also, at this time a second set of PROMs was programed and checked. It was determined that in order to completely test data flow and protocol handling an extensive test apparatus would have to be constructed. This test module is described in the next section.

B. TEST MODULE

The design considerations that led to the construction of the test module were requirements that all data lines, control lines, and flags must be able to be continuously monitored as well as simulated, i. e., inputted by hand toggled switches. Figure 6 shows the configuration employed. Figures 7, 8 and 9 show wiring details of the test module. This design allows the components to be tested singly as well as interconnected. Testing proceeded on the host adaptor and its MC, followed by the RI and its MC, and finally, testing of the units together. Figure 10 shows the schematics of the interconnections between the HA, RI, and Buffer Board. The Buffer Board acts as an interface between the test module and the unit being tested. This allows monitoring of all signals without causing signal drain. The physical layout of the Buffer Board is shown on Figure 11. Test procedures will now be described.



TEST MODULE- Consists of the following units:

POWER PANEL- Supplies 5 volts, -9 volts, and a common ground.

STATUS PANEL- Contains display lights which monitor the status of all data and control lines to, and from, the RI and HA.

SIMULATION PANEL- Contains switches that allow simulating inputs of all data and control lines into the RI and HA.

CLOCK 1- Supplies external clock input (simulating transmit clock) to the two micro-controllers.

CLOCK 2- Supplies external clock input.(simulating transmit clock) Required to be at least four times slower than clock 1. Supplies clocking to the RI.

DISPLAY PANEL- A 32 light display panel which is used to monitor the program counters in each MC. Used as a debugging tool.

FIGURE 6. Test Module Configuration.

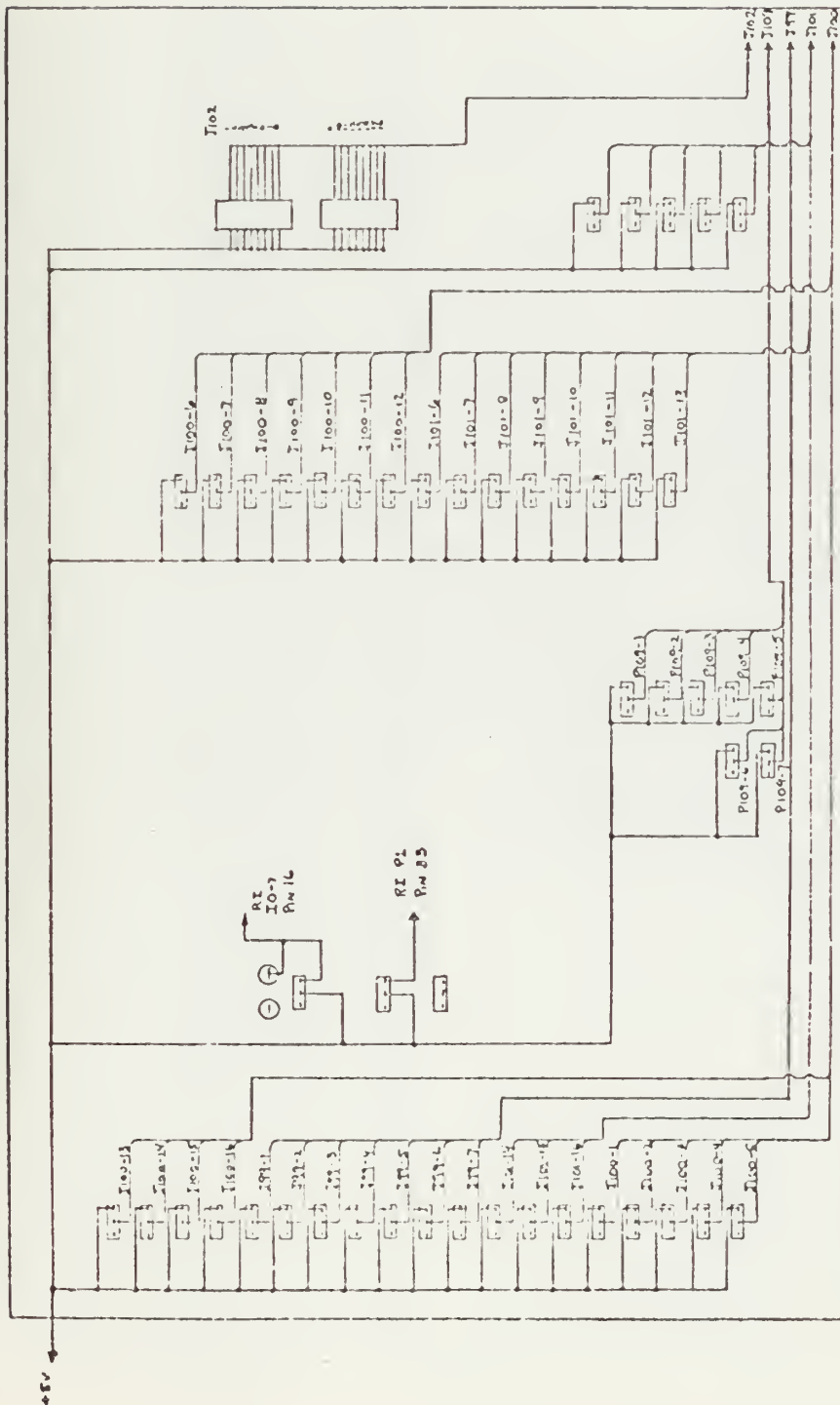


FIGURE 8. Simulation Panel Wiring Diagram

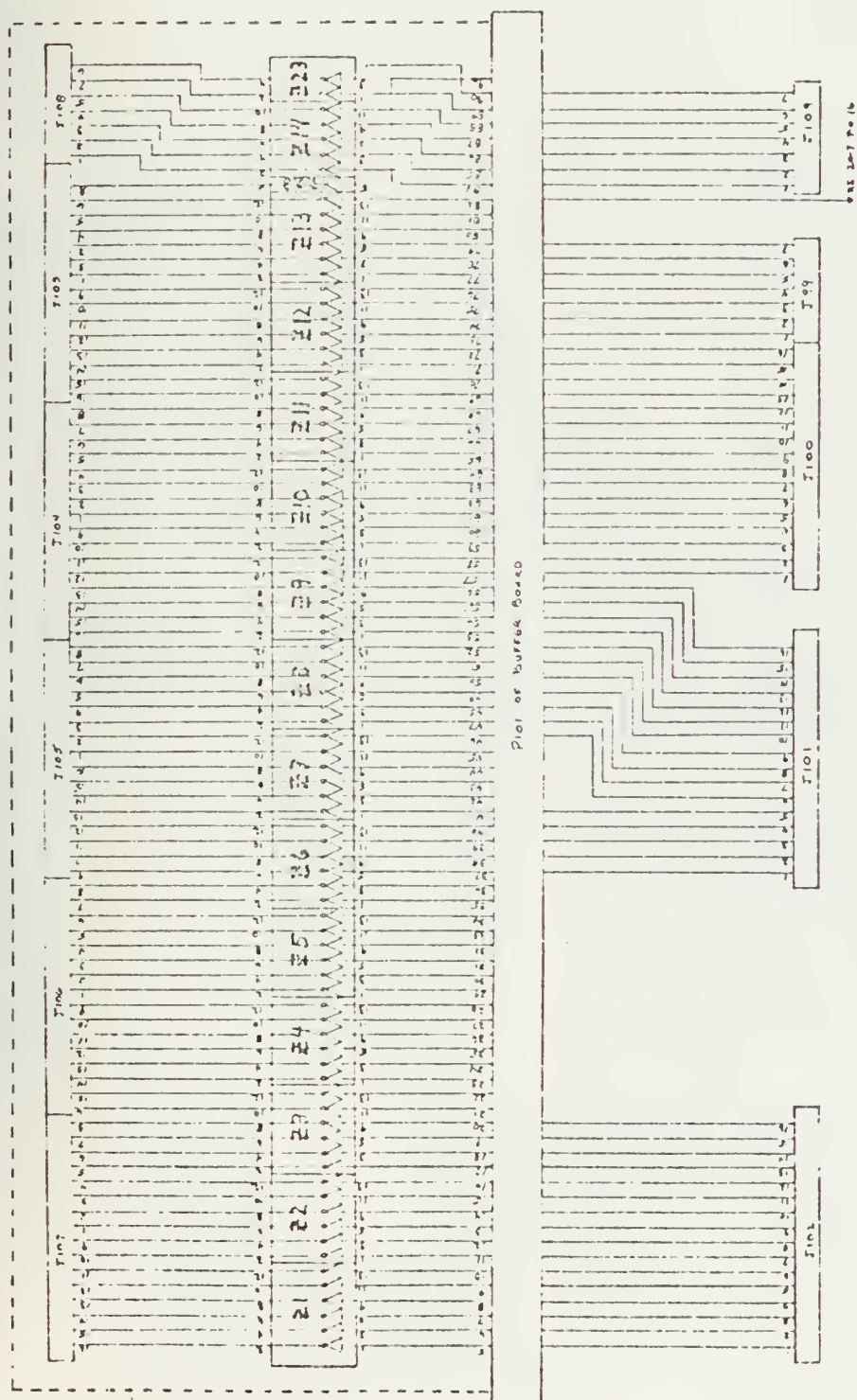


FIGURE 9. Status Panel, Simulation Panel, and Buffer Board Interconnecting Wiring Diagram.

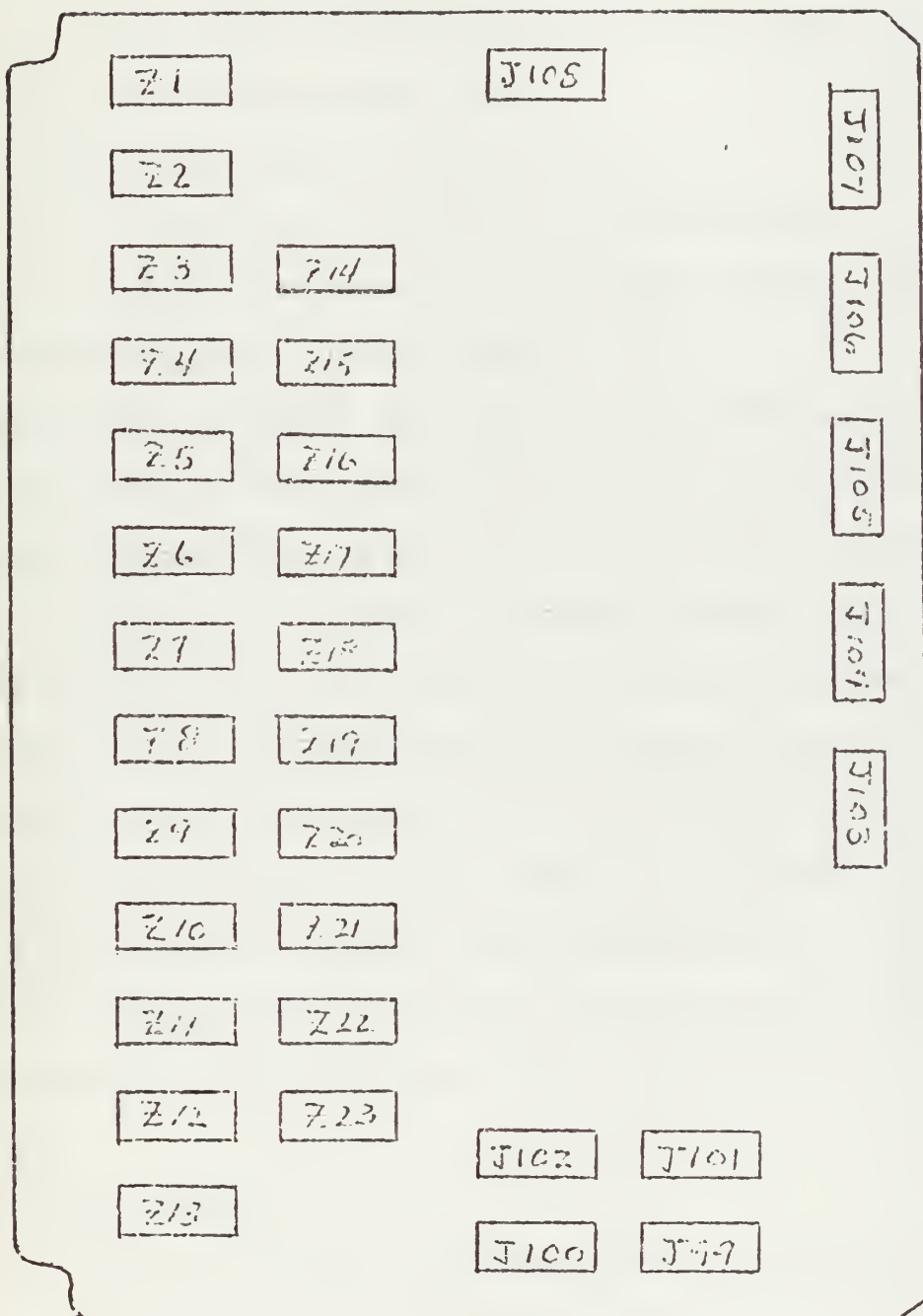


FIGURE 11. Buffer Board Physical Layout.

1. Host Adapter Test Setup

To test the HA singly, the following procedure must be followed:

- a. Buffer Board plugs installed: P99, P100, P101, P102, P103, P104, P105, P106, P107.
- b. All host adapter switches in "normal".
- c. All ring interface switches in "simulate initial".
- d. All PDA switches to "off". All other switches on the simulation panel are disconnected.

This setup allows simulating all outputs from the PDA and the RI, while monitoring all output data and control lines from the HA. After supplying the required power and clock inputs the HIA is ready to be tested. A complete test consists of forcing the HIA through all paths in the flowcharts of the receive and transmit sequences listed in Appendix C. In the transmit and receive sequences simulated data can be written into and removed from the FIFO buffer with the appropriate handshaking techniques. Tests accomplished have shown that the HA recognizes, decodes, and properly passes all local command messages. Simulated data bytes flow through the FIFO buffer correctly in both directions, and the micro-controller functions correctly.

2. Ring Interface

To test the RI singly the following procedure must be followed:

- a. Buffer board plugs installed: P99, P100, P101, P103, P104, P105, P108, P109 and RIMC.
- b. All HIA switches in "simulate initial".

- c. All RI switches in "normal".
- d. Data in switch "off".
- e. All RI internal switches "off". All other switches are disconnected and position doesn't matter.

Again the testing of the unit is accomplished by following the flowcharts (listed in Appendix B) to exercise all functions of the RI micro-controller. In testing the RI it was much more difficult to determine if data was flowing and being encoded correctly. Two factors caused this difficulty; serial data flow to and from the ring, and the fact that the repeater has not yet been constructed. This problem area is discussed in more detail in section VI.

3. Ring Interface and Host Adaptor

The final test was to check the Node as a single entity. This was needed to prove that the protocols and handshaking routines between the RI and HA actually work. The following procedure is to be followed to test the units together:

- a. Buffer board plugs to be connected: P102, P103, P104, P106, P107, P108, P109, RIMC.
- b. All PDA switches "off".
- c. All RI internal switches "off".
- d. Data-in switch "off". All other switches are disconnected.

The tests consisted of going through all receive and transmit sequences. As both micro-controllers are operating simultaneously it was found to be somewhat difficult to follow the internal handshaking, but with familiarity this problem lessened. All Local Command Messages were received by the RI correctly. However, it was impossible to check PNAME memory for correct writes and erases. This problem

still exists concerning the input of serial data, which will be resolved upon incorporation of the repeater.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. STATE OF THE PROJECT

At this time the NPS Data Communication Network has one working Node with the exception of a repeater. In addition, a second Node has been constructed, but not tested due to lack of operable integrated circuit components. The complete Node has been thoroughly tested at hand clocked speeds. This testing included functional testing of the host adapter and its micro-controller singly. Each function of the micro-controller (see Appendix C) was exercised. Simulated data words were inputted from both the PDA and the RI side to insure proper operation of the FIFO buffer. Using the flowcharts listed in Appendix B the RI was similarly exercised. Using slow speeds, outputted serial tokens were visually checked for correctness. Finally, the RI and HA were connected via the proper control and data lines, and the Node was tested as a functional unit. Of special interest was whether the internal protocols would work in this total environment.

Simulation of the PDA was accomplished via the Test Module. Local command messages as well as regular transmit messages were simulated to test all handshaking routines. All protocols functioned as designed. The following minor problems were encountered during this testing:

1. Power fluctuations

The test module, and all components of the Node are supplied by the same power source. The stability of the 5 volt power supply was

found to be critical. Fluctuations above 5.5 volts or below approximately 4.5 volts caused the micro-controller programs to become erratic. Ensuring good connections and proper sized wiring solved this problem, but a regulated power supply is recommended for future work.

2. Integrated Circuit Chips Inoperable

Several ICs were found to be faulty on the prototype Node. The 74161 synchronous 4-bit counters, used as program counters, were found to be especially susceptible to failure. The ability to test the RI and HA singly was a powerful debugging tool. The Display Panel, which allowed monitoring of the MC program counters, was valuable, in that it allowed a constant check on MC activity.

3. Lack of Repeater

The fact that the Repeater had not been constructed caused testing problems. The transmit clock was simulated with two external clock inputs. It was found more difficult to simulate the "recovered" clock, used during receive sequences. This fact also made it impossible to simulate known data streams from the ring side of the RI. This prevented testing the decoding and PNAME matching functions of the RI.

4. Inability to Test at High Speed

The external clock inputs were varied from 1 to 100 kc during testing. While no problems were encountered at the higher cycle speeds, no real high speed test results can be assumed. Because of the test procedures employed, the micro-controllers were spending the vast majority of time waiting for the next hand simulated input, so no conclusions can be drawn about high speed operation. Design

speed testing should be performed when a high speed simulated PDA can be connected to the Node.

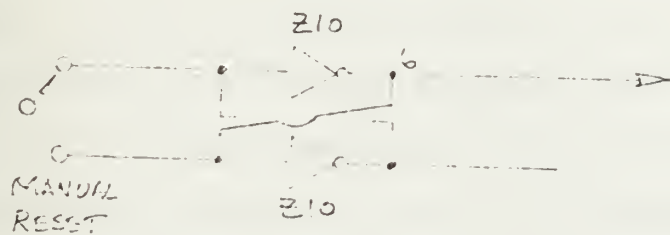
5. No Reset Line Into the RI Micro-controller

On the prototype Node, hand toggled reset switches were provided to restart both micro-controllers. A reset line is provided from the HA to the RIMC, and is enabled as one of the local command messages from the PDA. The required circuitry was not incorporated into the design or implementation of the RI micro-controller. Figure 12 shows the circuitry modification required to include this function.

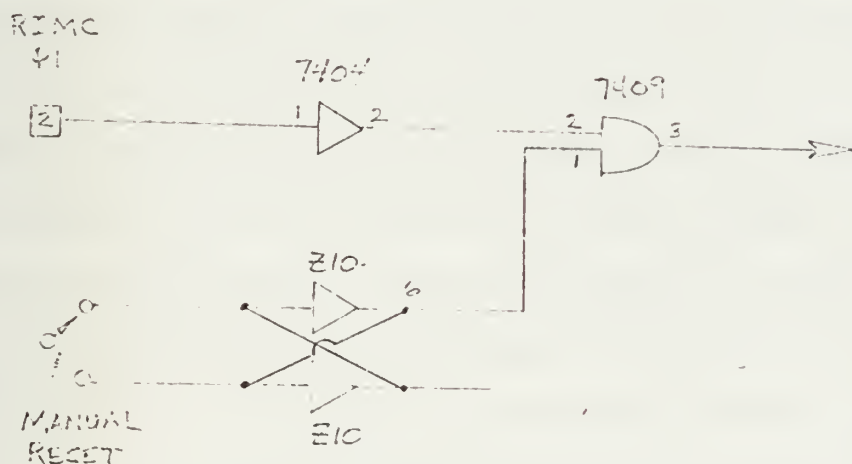
B. RECOMMENDATIONS

Recommendations on future development of the project deal with two areas: first, a recommended hardware modification; and second, follow-on test requirements.

After studying the various host processors at NPS, it is considered that buffering of data between the RI and the hosts is a common requirement. As the FIFO buffer would then be common to each Node, a logical hardware modification would be to shift the buffering function from the HA to the RI. This would be in keeping with the basic design concept that the ring interface is host processor independent, and provides functions common at each Node. This modification would require both hardware and software changes to the RI and HA. Handshaking techniques between the HA and RI, as well as between the HA and PDA, would remain the same. The FIFO buffer circuitry would be an integral part of the RI. During a receive sequence the RI would still signal the host concerning the incoming message, but data bytes would immediately start filling the buffer. The functions of the HA would be reduced to simply restructuring the data format



EXISTING CIRCUIT



PROPOSED CIRCUIT

FIGURE 12. Reset Line Modification To RI Micro-controller

from the 8 bit parallel format of the RI to the format required by the host, and providing the necessary control and status lines. This modification would allow the direct connection of some hosts (such as a mini-computer) to the RI without the need for a host adaptor, with handshaking procedures accomplished using software.

Of immediate necessity is construction of the Repeater, as high speed testing requires this hardware in the system. The second Node should then be completed and tested in order to constitute a two Node network. At this point high speed tests can commence. Because of equipment availability the use of mini-computers for initial testing is recommended. The use of one mini-computer would allow all LCMs to be tested, and would also allow data to be transmitted Node to Node. Maintaining data streams of less than 1024 bytes in length would allow the Test Module to be used as the second, or receiving Node. With software simulation of the PDA, the mini-computer could transmit a known data stream to the second Node. The Test Module would allow connection of the second Node to the ring. Very short data streams could also be visually checked for accuracy by using the Test Module to simulate a receiving PDA and removing the data from the buffer. Message Transmission, and reception by the RI would be accomplished, however, at design speeds. After transmitting a message the software program could also simulate the receive sequence of the PDA and therefore status information, decoded from message status bits by the RI, would be automatically passed on by the RI after the return of the just-completed message. The status information could then be displayed or printed out on a teletype. This test procedure would fully check all functions at operating speeds.

Follow-on testing would be concerned with establishing two host processors on-line to check transmission of longer, more realistic messages. Two other projects of interest are designing software protocols for using the ring and an investigation into modifying the host adaptor to interface with other computer hardware.

In conclusion, the NPS ring data communication system is considered to be a viable means of sharing resources among computer systems. The data transmission rates are seen to be adequate with very reasonable costs-per-Node expected. Further developmental work is recommended with emphasis on the high speed testing, and applications analysis.

APPENDIX A

COMMON ABBREVIATIONS AND TERMS

- APN - Alter process name control line from the host adaptor to the ring interface. Called ALTER in Ref. 3. When enabled by the HA it signals the RI to either write into or erase from PNAME memory, depending on the state of the WTNM control line.
- BPV - Bipolar violation error. An error condition which occurs when three or more "one" bits are detected in a row in the serial data stream. This condition does not normally occur due to the encoding of data bits. This encoding is shown in Figure 3. The BPV is also used to alert the RI to the start of a possible token.
- CTL - Control token. A recognizable token or code byte, which is used to maintain and pass control of the ring network.
- DCTD - The disconnected flag from the RI to the HA. When enabled this flag signals the host that the Node is not connected to the ring. The same signal the RI used to set this flag is also used to signal the repeater, which actually performs the function of electronically disconnecting the Node from the ring.
- DEM - Demand control line from the HA to the PDA. In a receive sequence, enabling this line tells the PDA that the next word is ready. During a transmit sequence this line indicates that the last data word was accepted by the HA.

- DISC - Disconnect control line from the HIA to the RI. When enabled this line signals the RI to disconnect from the ring. This line is raised by the HA in response to a Local Command Message.
- DTL - Data Transmission line. The actual shielded cable that connects the Nodes. Proposed cabling is discussed by Brubaker in Ref. 2.
- EOF - End of file control line from the HA to the PDA. Enabling this line indicates that the HA has completed its operation. During a receive operation this line indicates an error was detected on the incoming message.
- EOM - End of message token. A message delimiting token or code that signals a RI that data bytes in the message are finished.
- EOR - End of record control line from the HA to the PDA. Enabling this line indicates that the HA has completed its operation. A normal ending to a receive sequence.
- HA - Host Adaptor. A functional hardware unit which acts as a buffer and adaptor between the IBM 360 and the RI. Also an acronym for the host accept control line from the host adaptor to the ring interface. Called HACCEPT in Ref. 3. During a receive sequence this line signals to the RI reception of a data byte by the host adaptor.
- HDR - Host data ready control line from the HA to the RI. Called HDATARDY in Ref. 3. Enabling this line signals the RI that the next data byte is ready during a transmit sequence.

- INT - Interrupt control line from the HA to the PDA. Enabling this line signals the host to prepare to receive a message. The host must respond by jumping to the receive mode.
- LCM - Local command message. Messages sent from the host used to control the actions of the RI. LCMs are shown in detail in Figure 5.
- MSB - Message status bits. The last three bits in a message which are set by the receiving Node to indicate to the sending Node how the message was received. Figure 3 lists interpretations of the message status bits.
- NODE - As used in this paper a Node consists of a ring interface with its associated host processor or processors. As designed the ring can support up to 256 Nodes. This limit is primarily due to the addressability of PNAME memory.
- PDA - Parallel data adapter. An IBM hardware unit which provides interfacing of external devices into the IBM 360.
- PNAME - Process name memory. A 256 x 1 RAM which resides within the ring interface. Used to match any incoming message address byte to determine if the message is addressed to a resident host.
- RCRC - Flag from RI to HA. When enabled this flag signals the HA that a CRC error was detected during a receive sequence.
- RCV - Receive control line from the RI to the HA. Called RECEIVAL in Ref. 3. The RI raises this line to signal the HA that an incoming message is being received (and

has been recognized as being addressed to a resident host of that Node). Raising of this line pre-empt's a transmission sequence.

- RDR - Receive data ready control line from RI to HA. Called RDATA RDY in Ref. 3. During a receive sequence this line is raised to signal the HA that the next byte of incoming data is ready to be sent to the HA.
- RESET - Reset control line from the HA to the RI. When this line is enabled (strobed for a minimum of 1.1 microseconds) it forces the RIMC to reset the program counter back to the "initial" location. Used to restart the RI after a lockup due to an error condition.
- RI - Ring Interface. A functional unit of hardware which allows a host processor to link up and use the ring network.
- RID - Ring interface demand. A control line from the RI to the HA. Called DEMAND in Ref. 3. This control line is used during a transmit sequence to signal the HA that the data byte was accepted.
- RIHAMC - Ring interface host adaptor micro-controller. Controls actions of the host adaptor functions.
- RIMC - Ring interface micro-controller. Controls actions of the ring interface functions.
- ROVER - Receive overrun flag from the RI to HA. When enabled indicates to the HA that a data overrun occurred during a message reception. This data overrun occurs when the FIFO buffer is filled because the host did not accept

data fast enough during a receive sequence.

- RPTR - Repeater. A functional unit of hardware in a Node whose primary purpose is to recover clocking information from incoming data bits, reshape, and act as a line signal booster to the ring.
- RR - Read ready control line from the PDA to the HA. When enabled this line signals the HA that the host is ready to accept the next two bytes of data.
- RRERROR- Receive ring error flag from the RI to HA. When enabled, signifies an error was detected on incoming data during a reception sequence.
- RS - Read select control line from the PDA to the HA. When enabled, this line signals the HA that the host is in the read sequence.
- SMAL - Symbolic Micro-controller Assembly Language. A special language developed by Assistant Professor Gary A. Kildall to program the micro-controller read-only memories. Ref. 6 describes this language.
- SOM - Start message token. Used to signal the various Nodes that a message follows, and alerts the RIs to prepare to check the next byte for a match with PNAME memory.
- TM - Transmit message. A normal message to be outputted to the ring. As sent from the host the first two bytes consist of the address of the target Node followed by the address of the sending host. This is followed by data bytes.

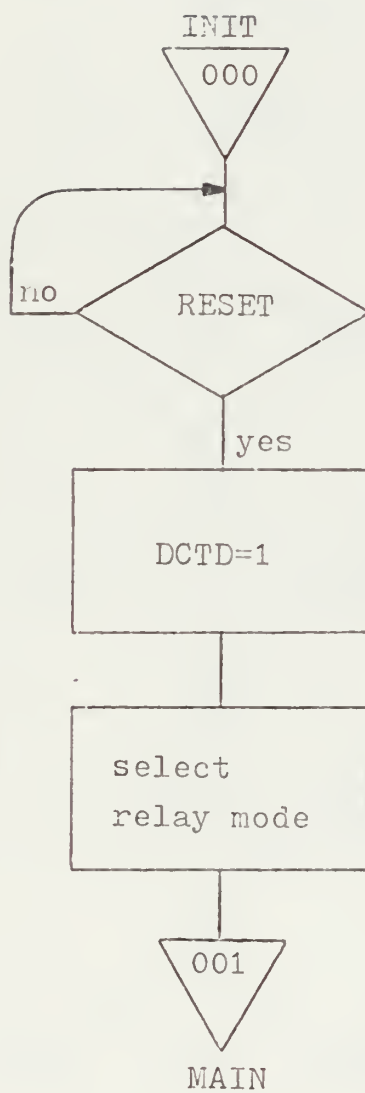
- TOKEN - A special recognizable byte that is used to control the ring and signal start and end of messages. All tokens start with three "1" bits which are interpreted as a bipolar violation and initiate a decoding sequence within the ring interface.
- WC=0 - Word count equals zero control line from the PDA to the HA. This line is enabled to indicate normal end of write operation, and in a read operation that the PDA will no longer accept data (an error condition).
- WR - Write ready control line from the PDA to the HA. When enabled this line signals the HA that the next 16 bits of data are ready to transmit from the host. Handshaking techniques can be followed by studying the flowcharts listed in Appendix B.
- WRTN - Write name control line from the HA to the RI. Called PNAME ACTIVE in Ref. 3. When raised simultaneously with an APN line it causes the RI to set the appropriate bit in PNAME memory.
- WS - Write select control line from the PDA to the HA. When enabled this line signals the HA that the Host desires to transmit a message.
- XCRC - Transmit CRC error flag from the RI to the HA. When enabled, signals the HA that MSB3 returned in a "one" state, which indicates that a CRC error was detected by a target RI during reception.
- XMIT - Transmit control line from the HA to the RI. When enabled this line signals the RI that the host desires to

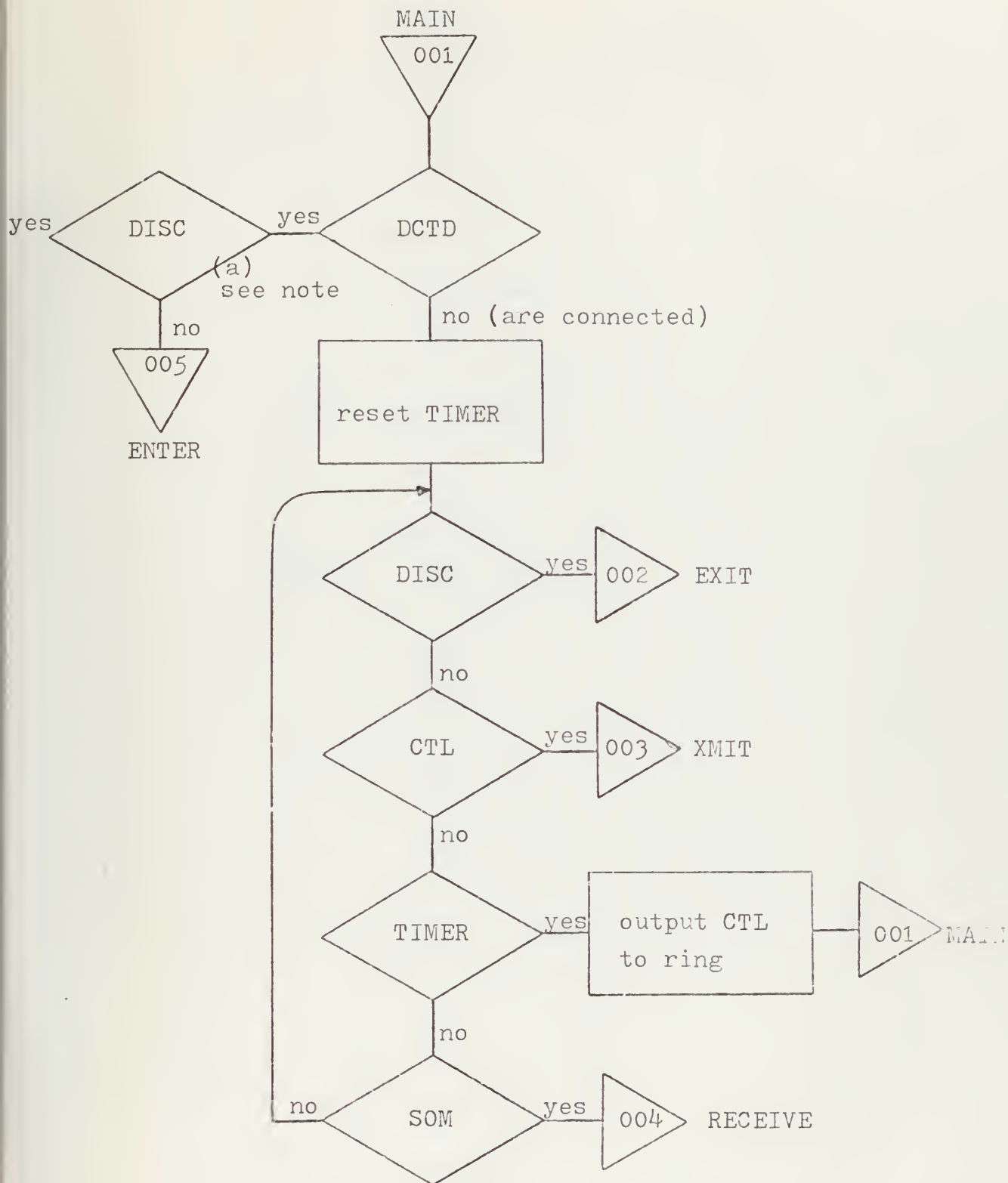
transmit a message to the ring. Called XMITL in Ref. 3.

XOVER - Transmit overrun flag from the RI to the HA that signals an overrun during a transmission sequence. It indicates that the host did not supply data fast enough to the HA to be transmitted to the ring.

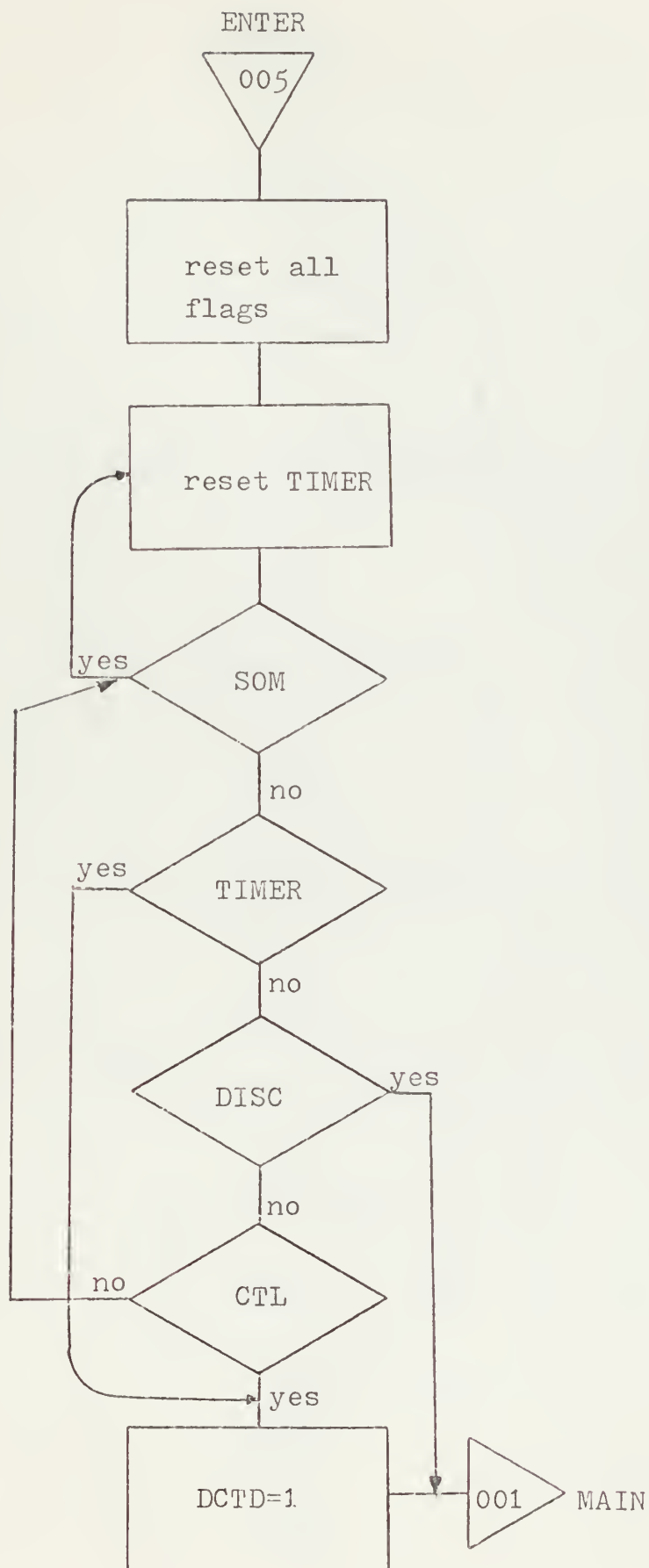
XRERROR- Transmit ring error flag from the RI to the HA. This flag signals an error during a transmission sequence.

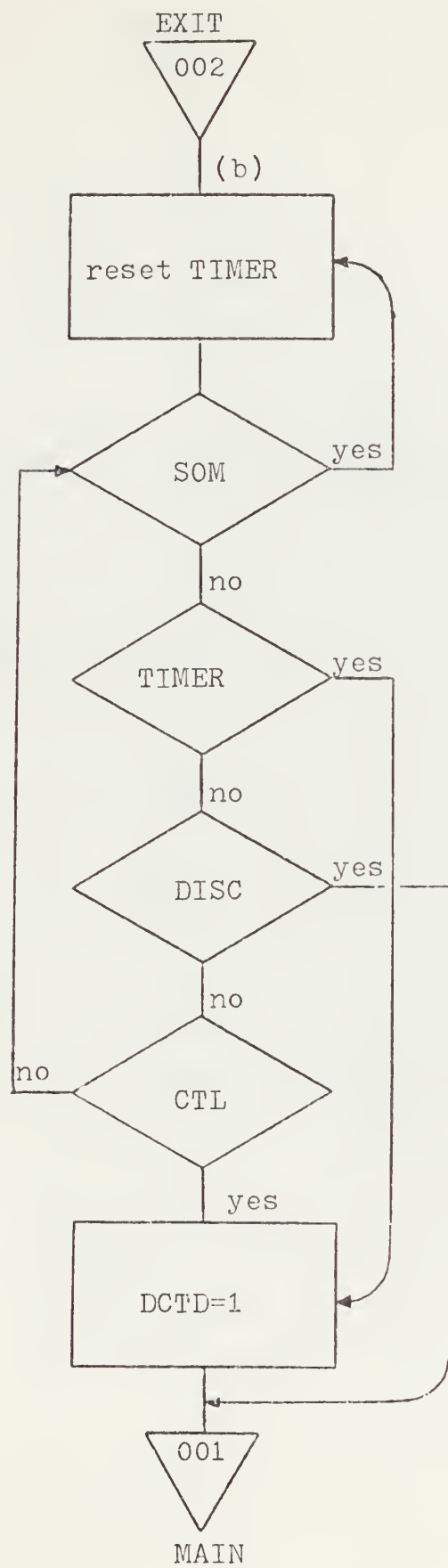
APPENDIX B
RI PROCEDURAL FLOWCHARTS

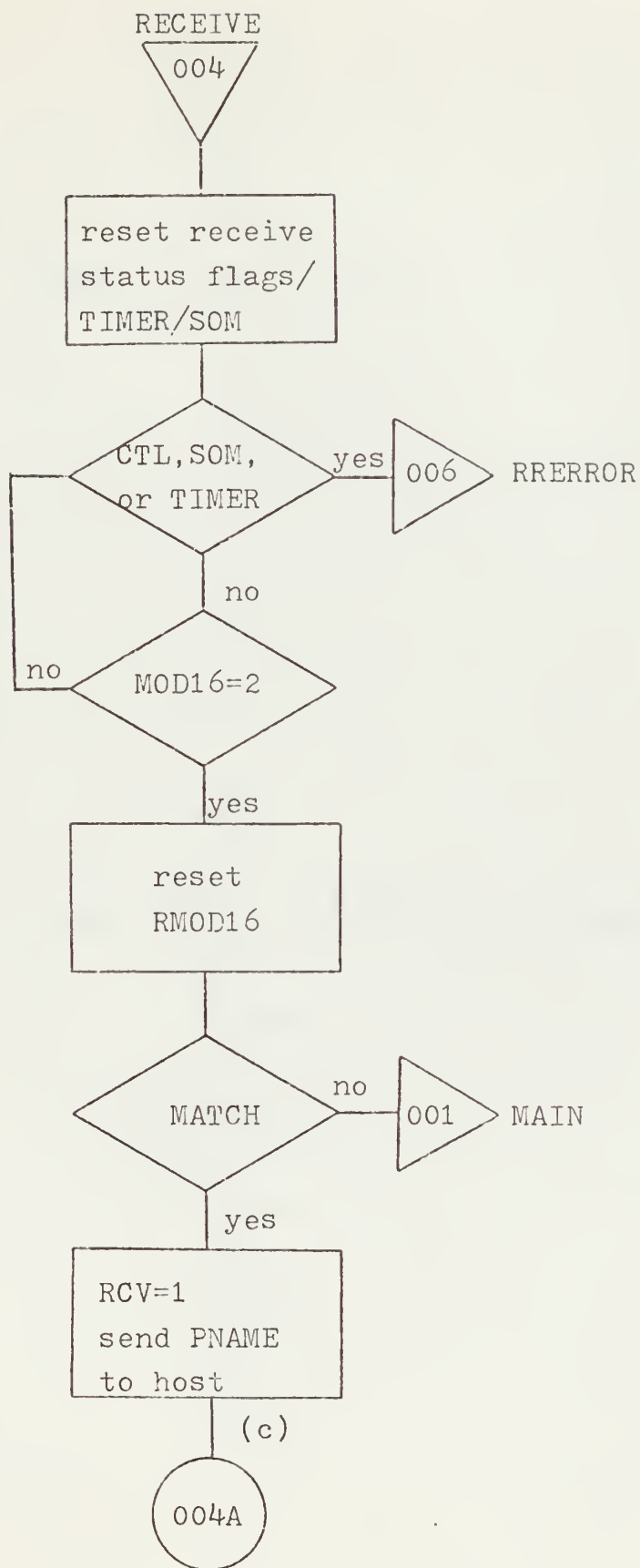


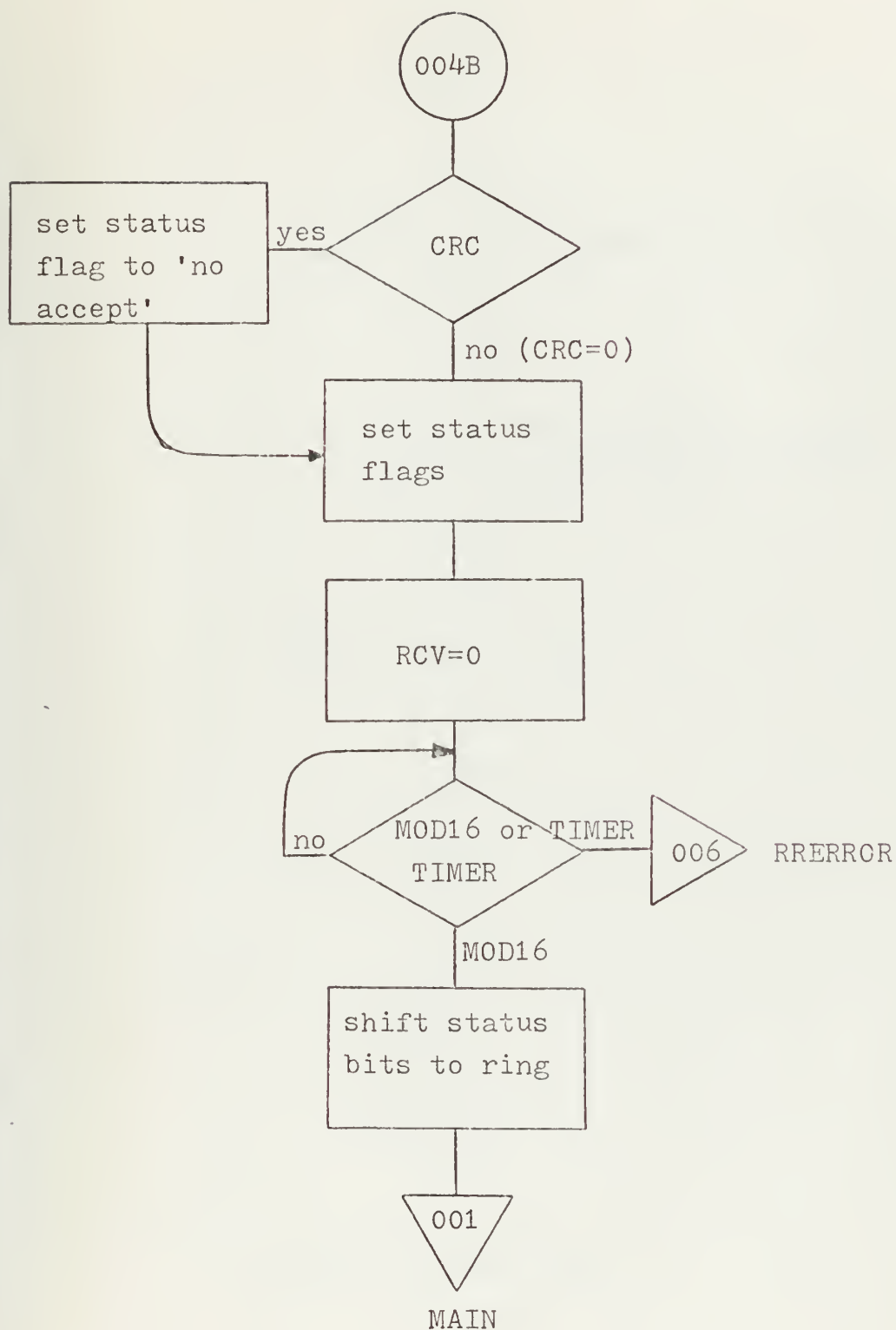


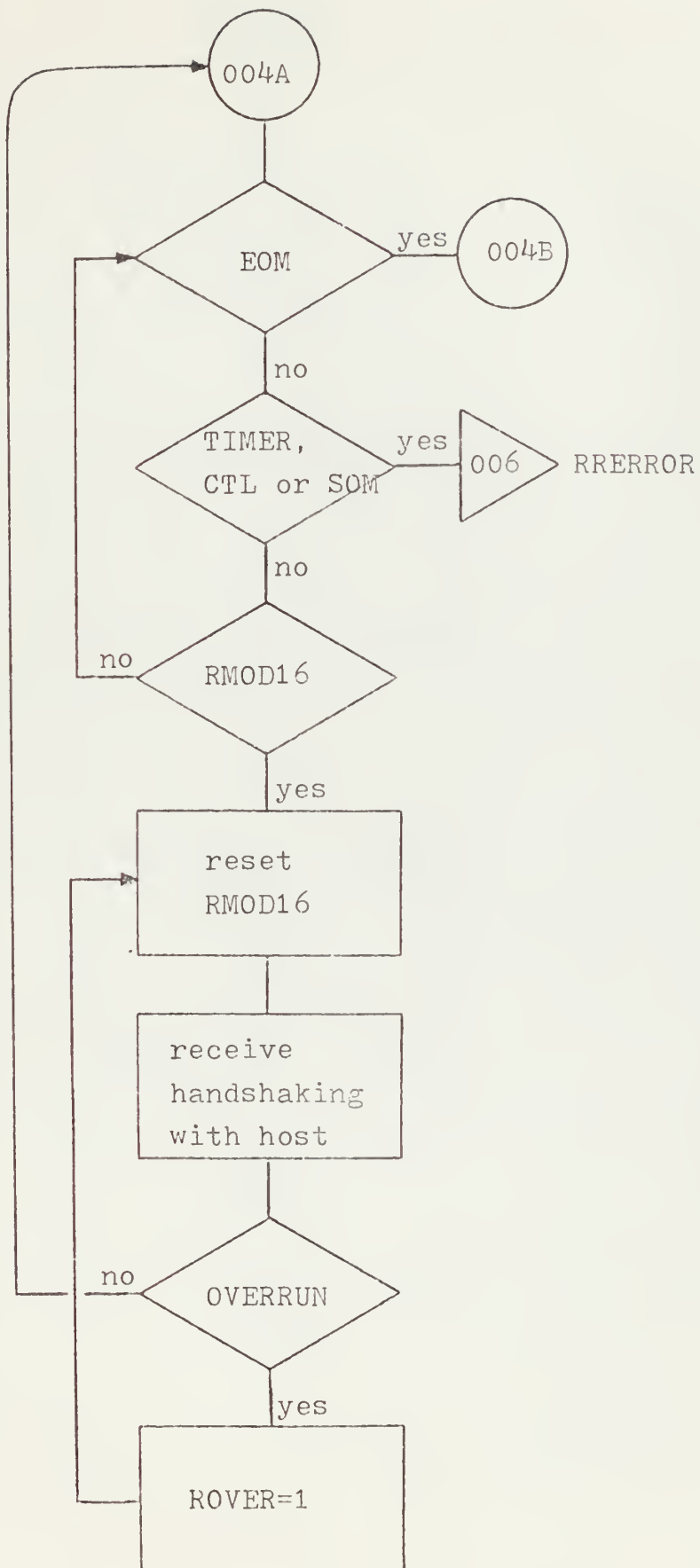
: the use of small letters, (a), is used to denote areas in the procedural flowcharts that can be correlated with functions, or s in the HA procedural flowcharts.

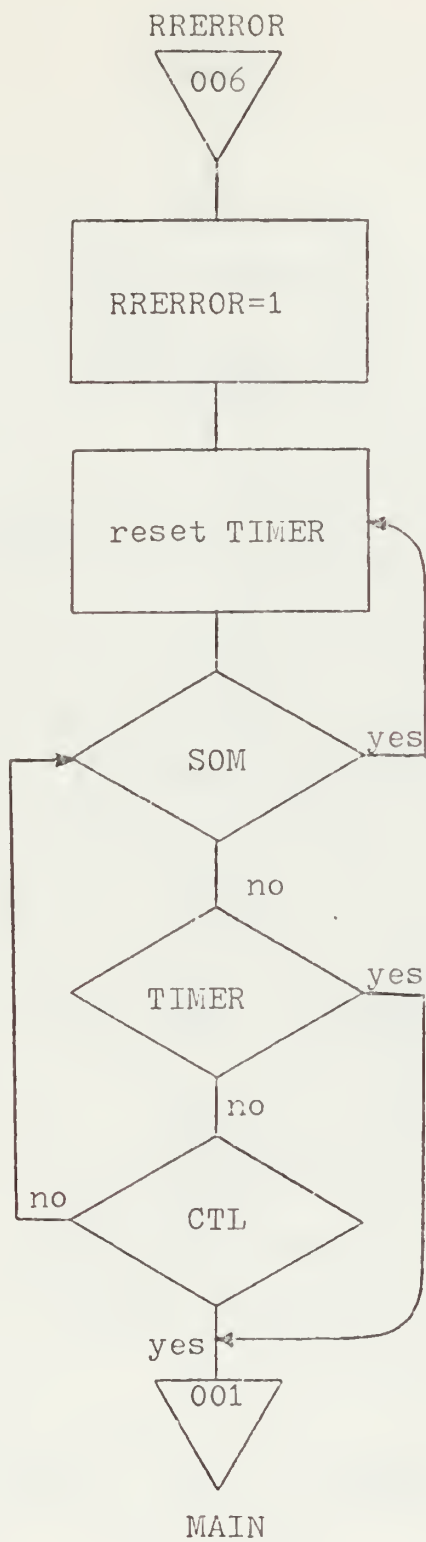


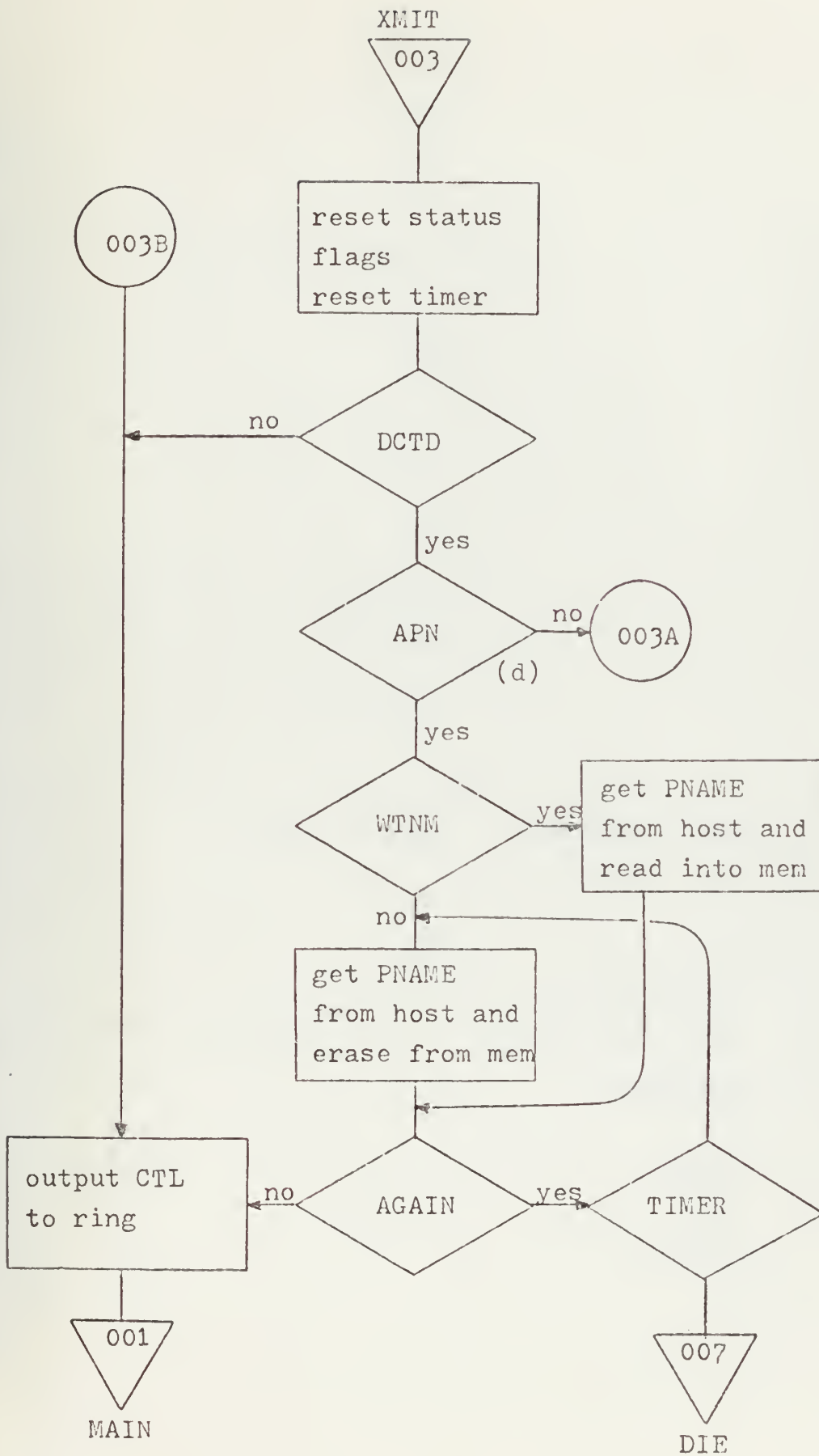


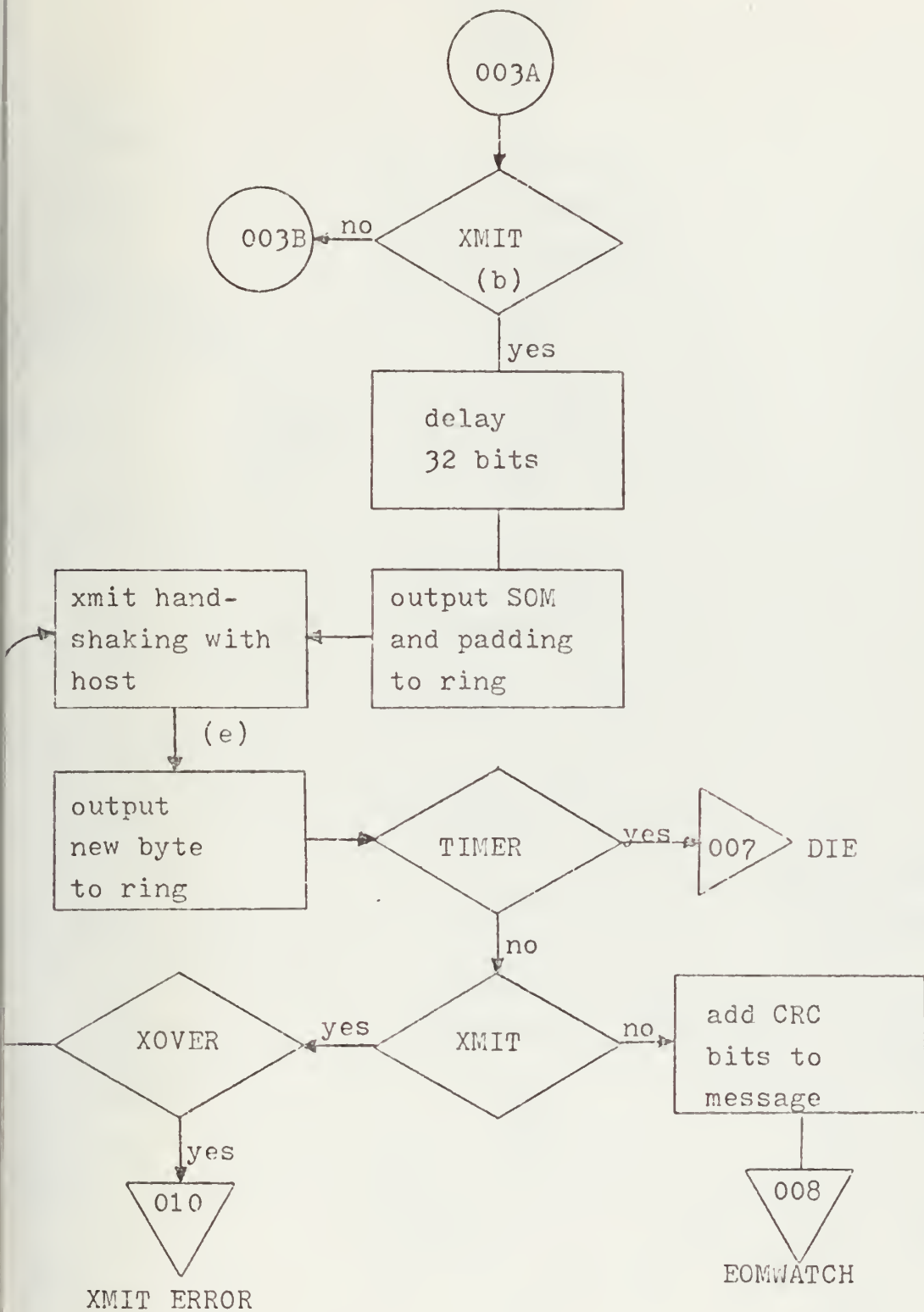


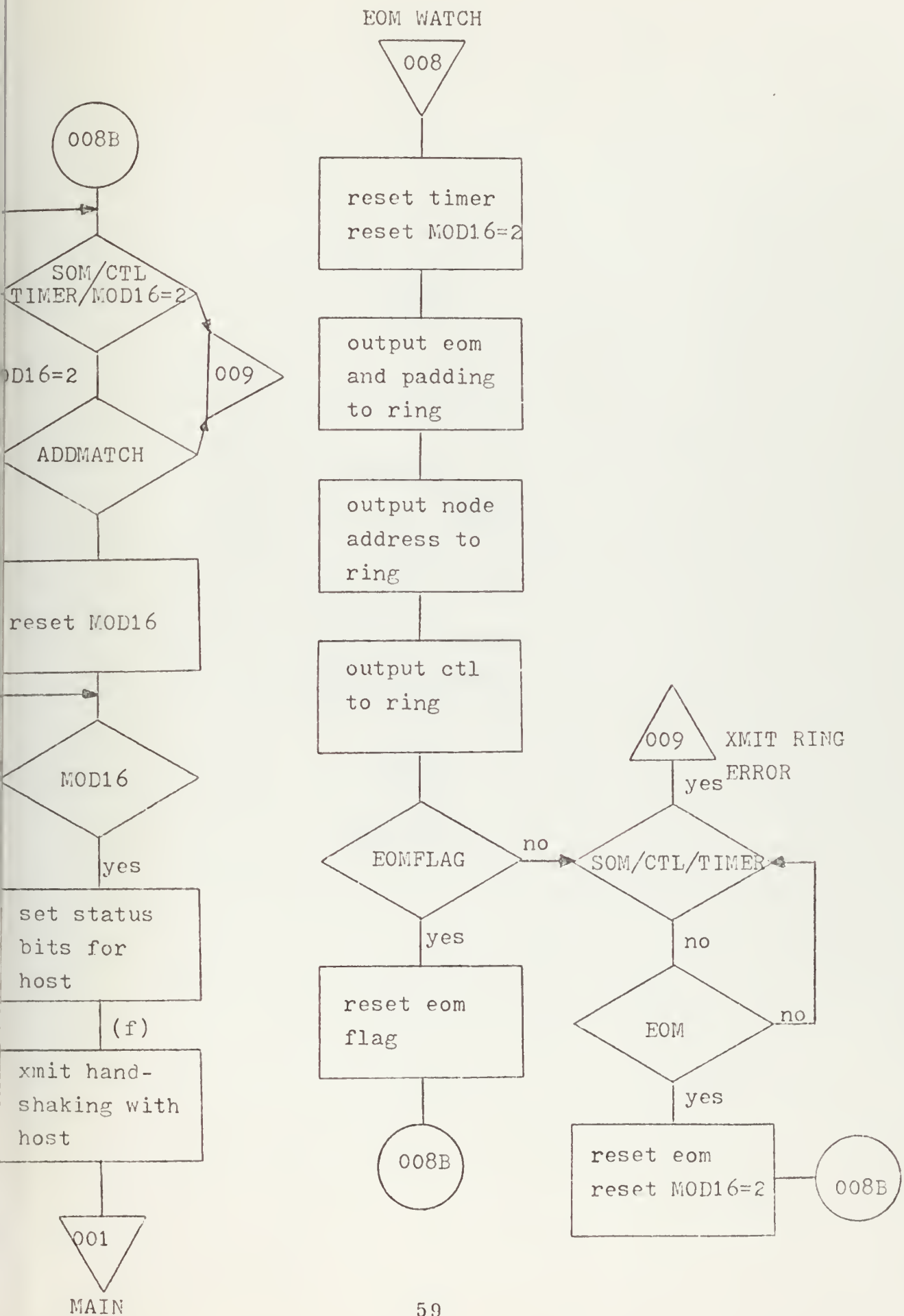


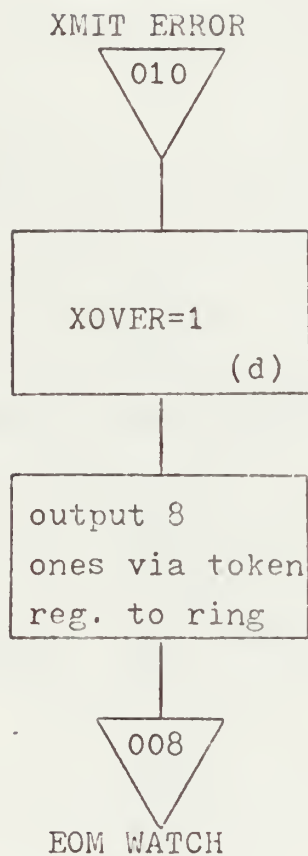


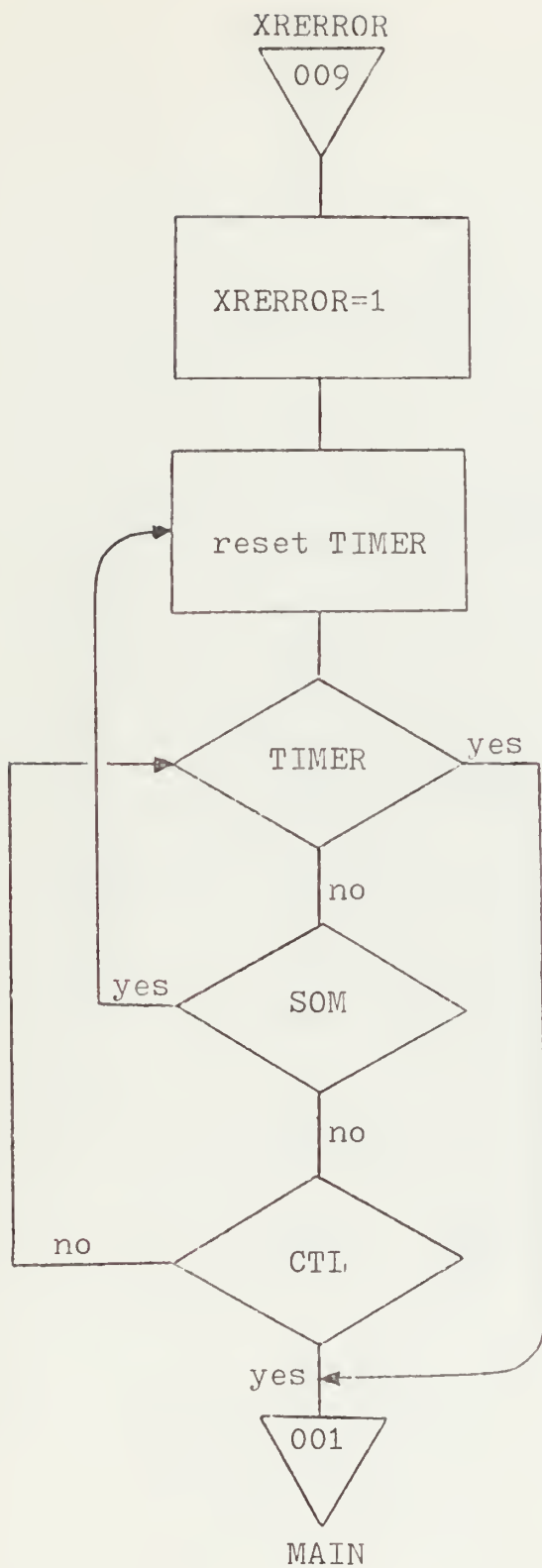


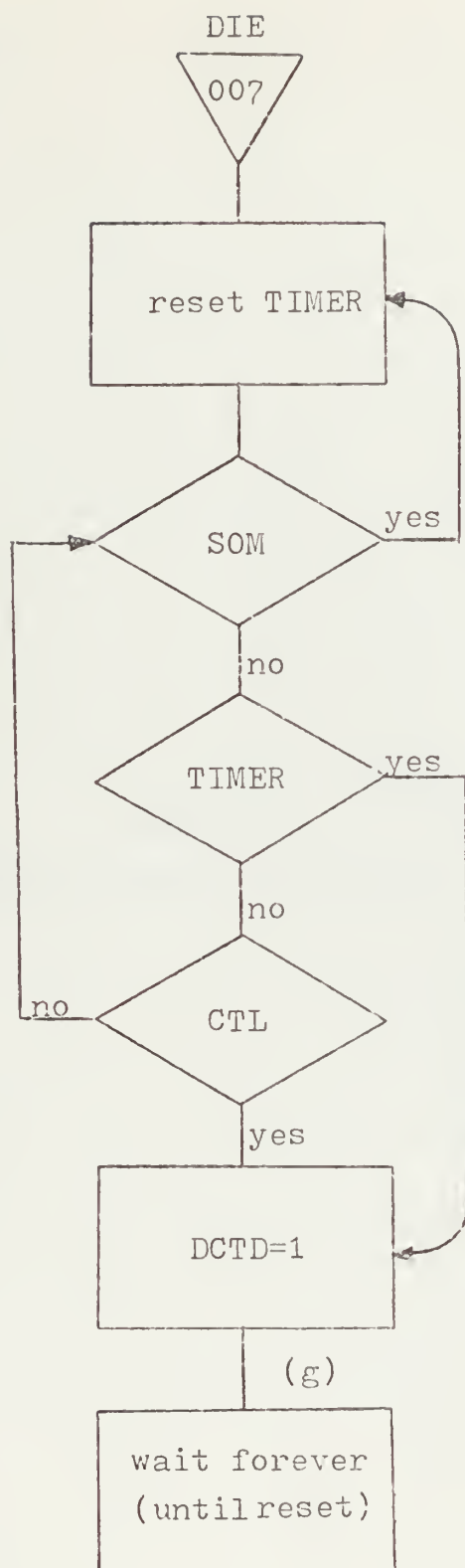




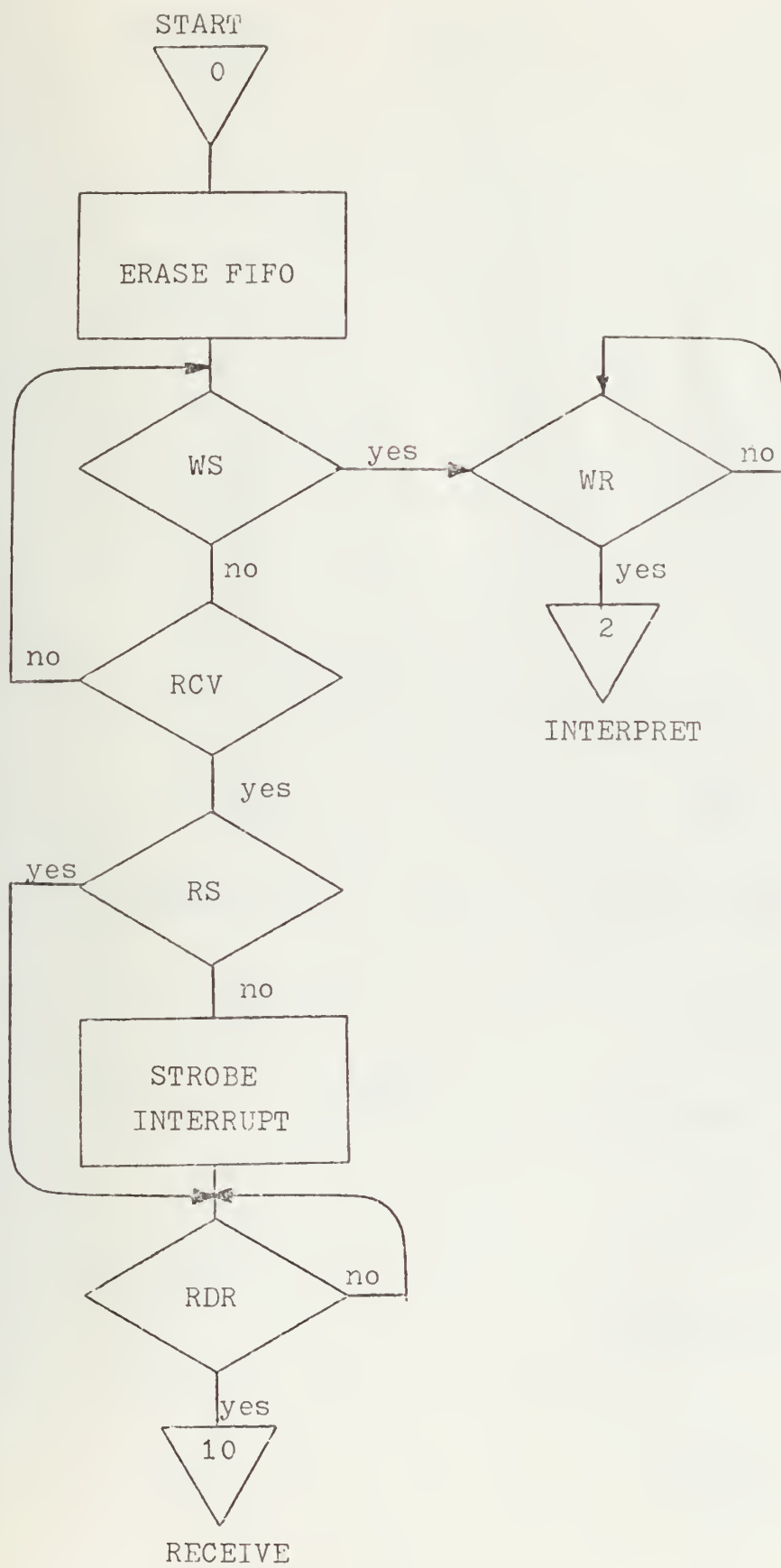


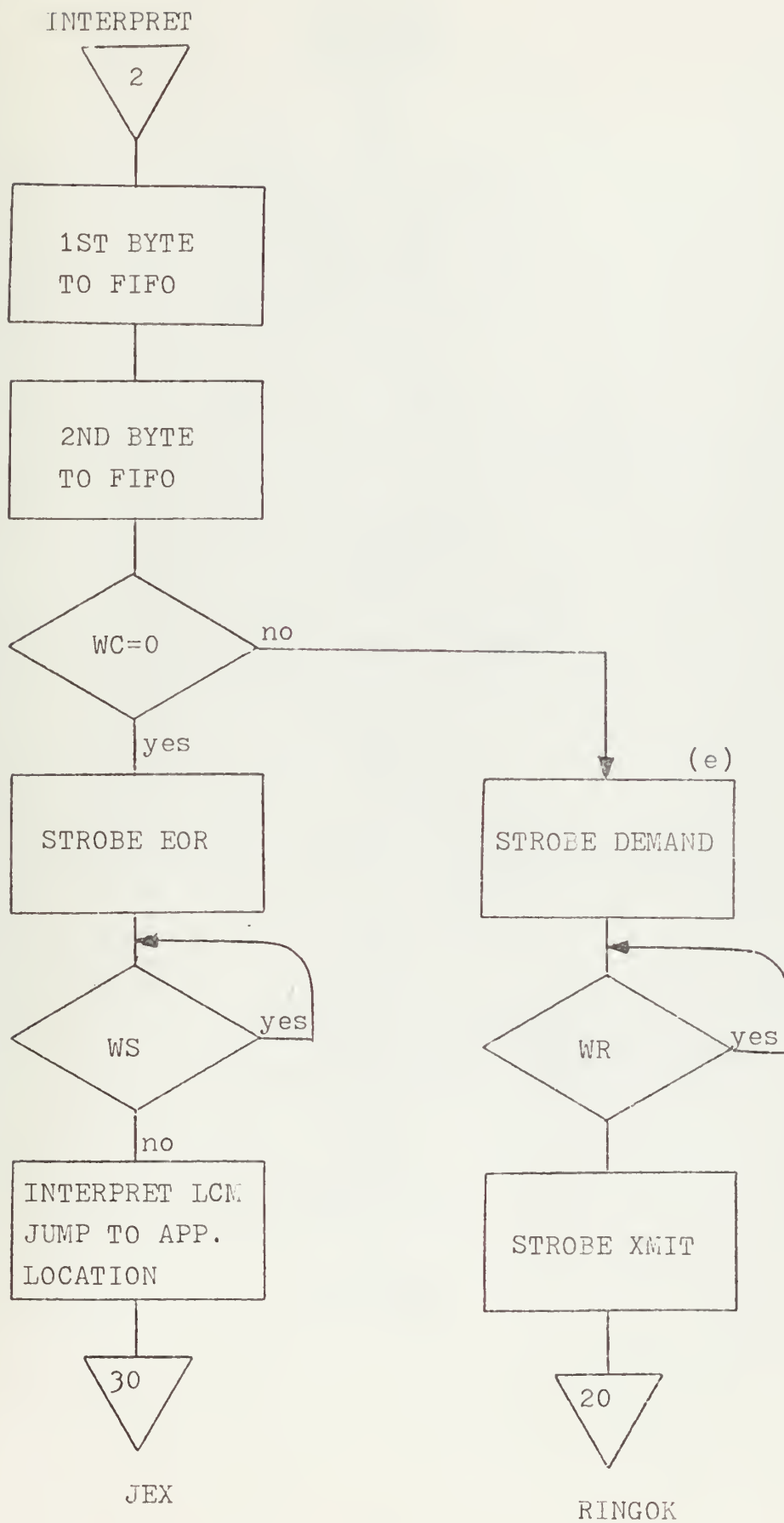


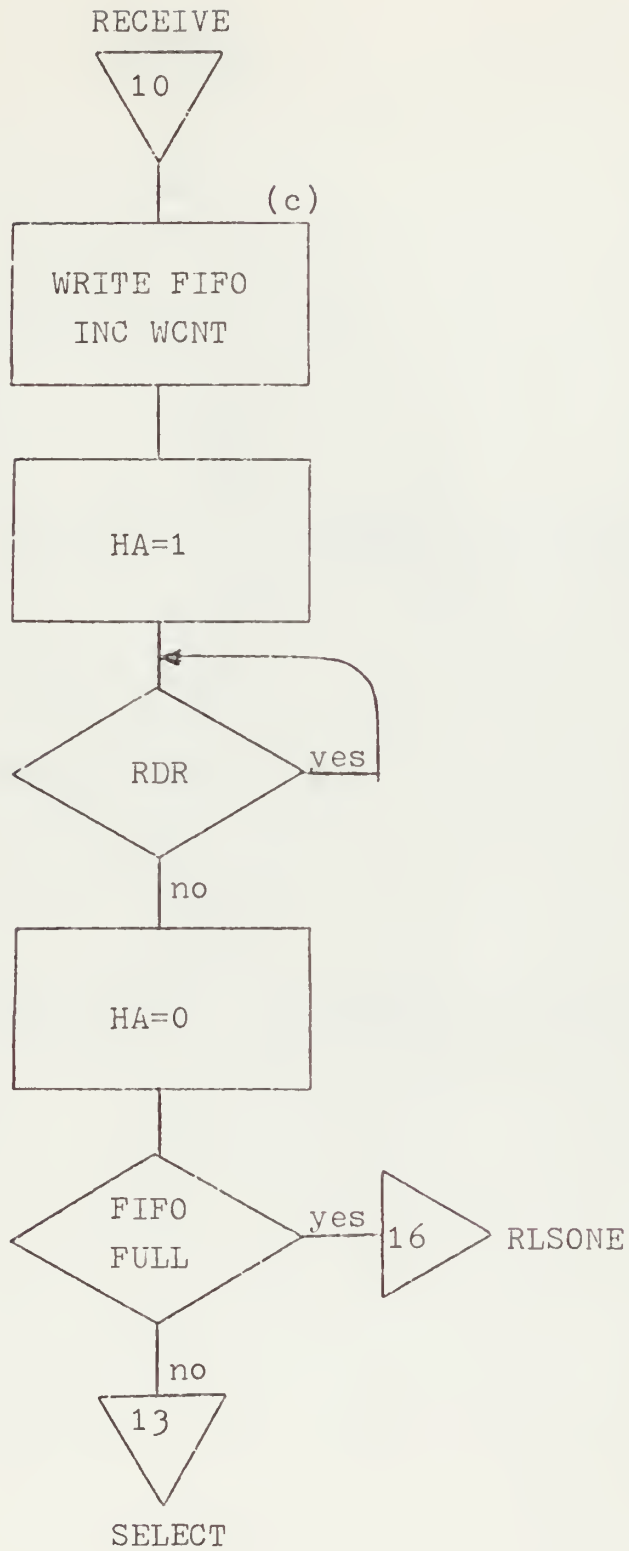


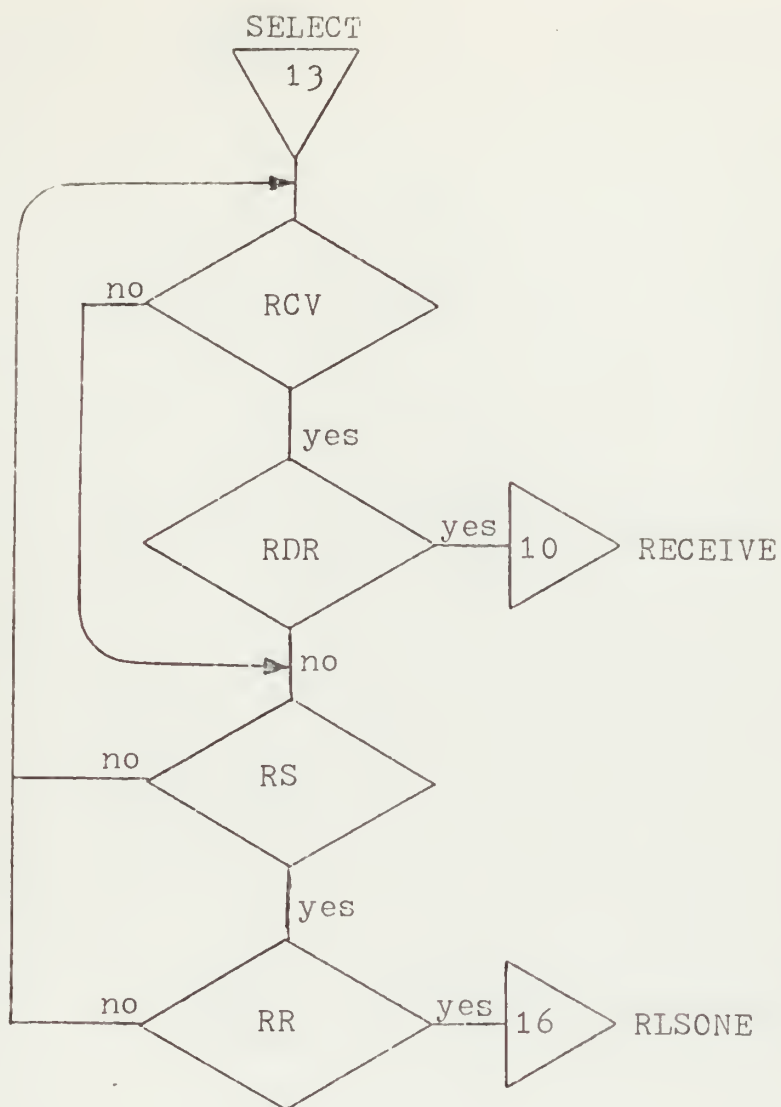


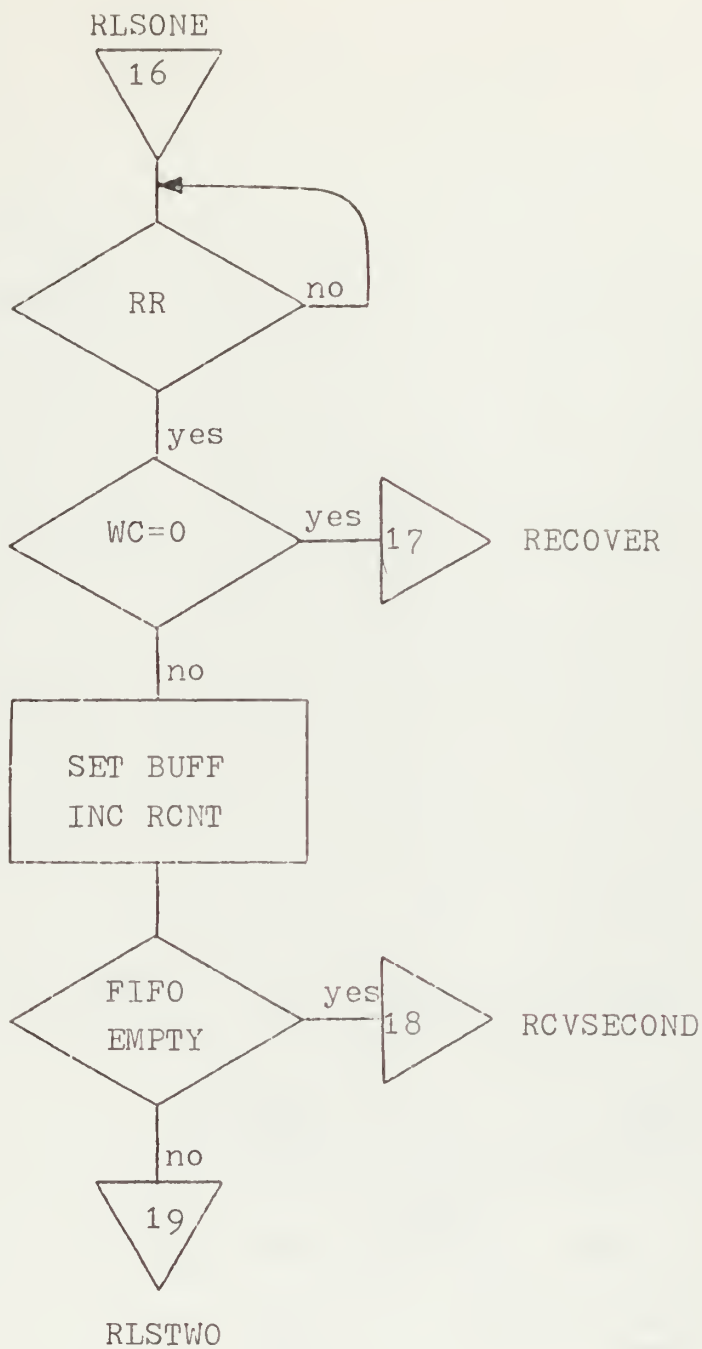
APPENDIX C
HA PROCEDURAL FLOWCHARTS

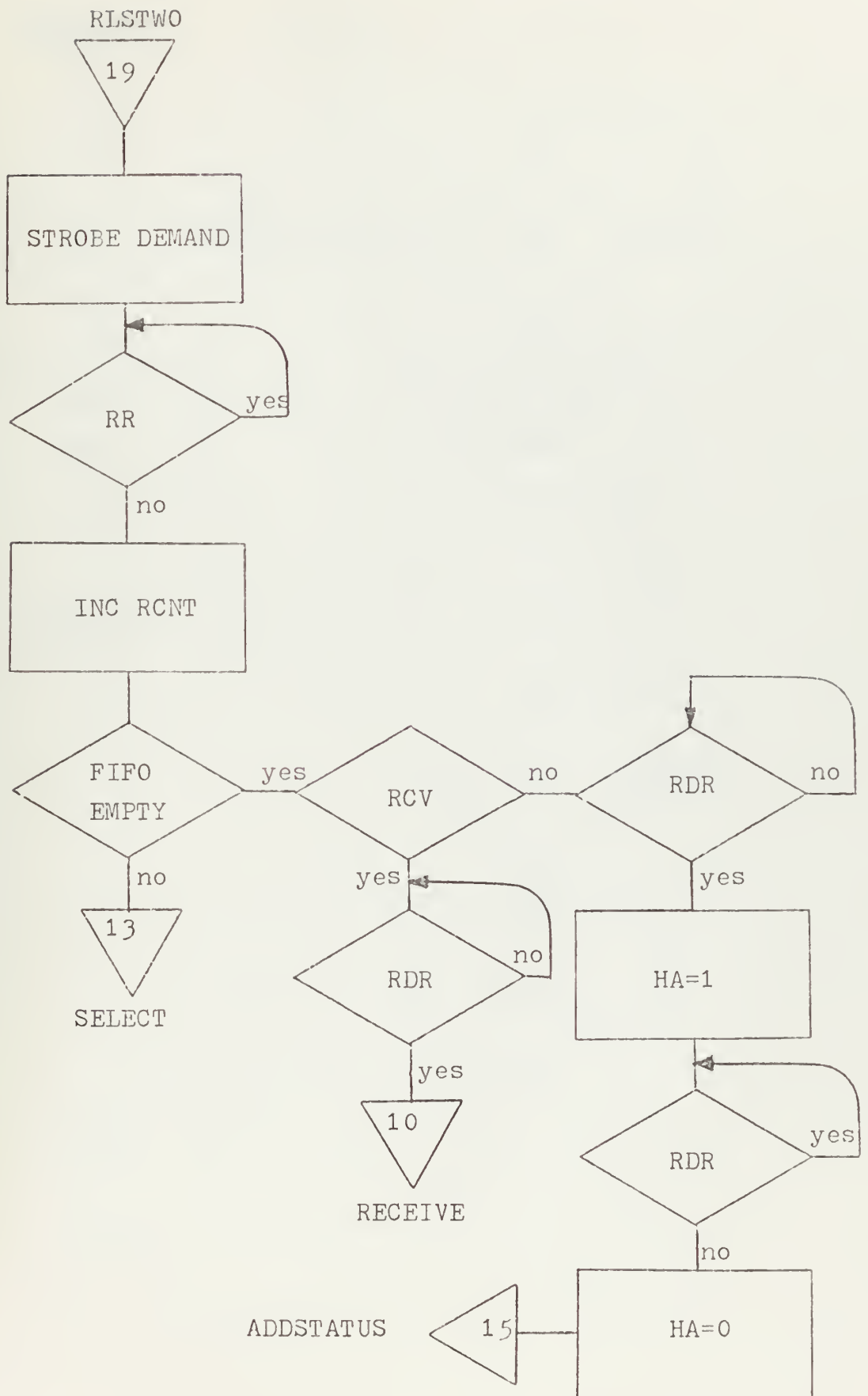


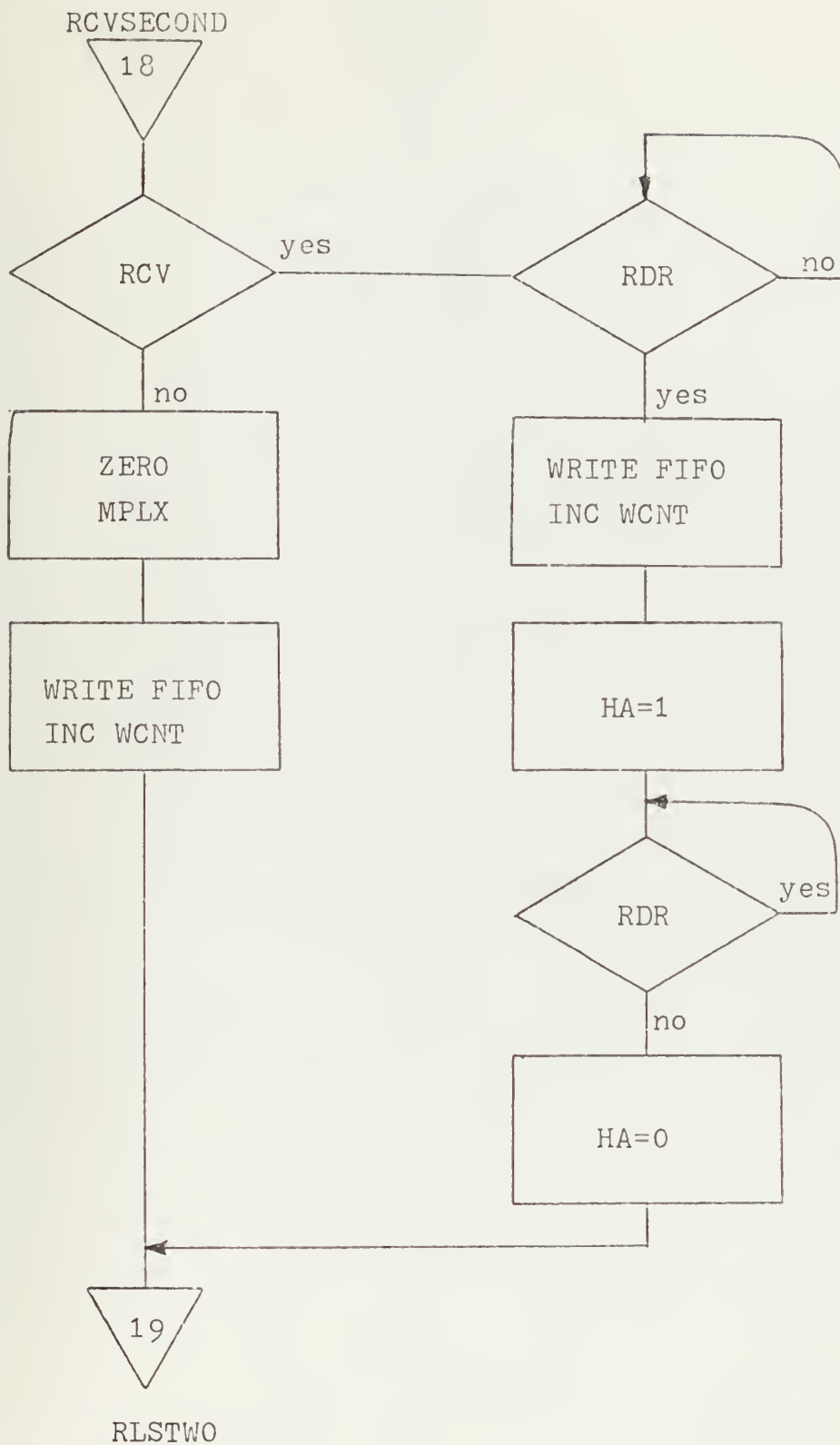


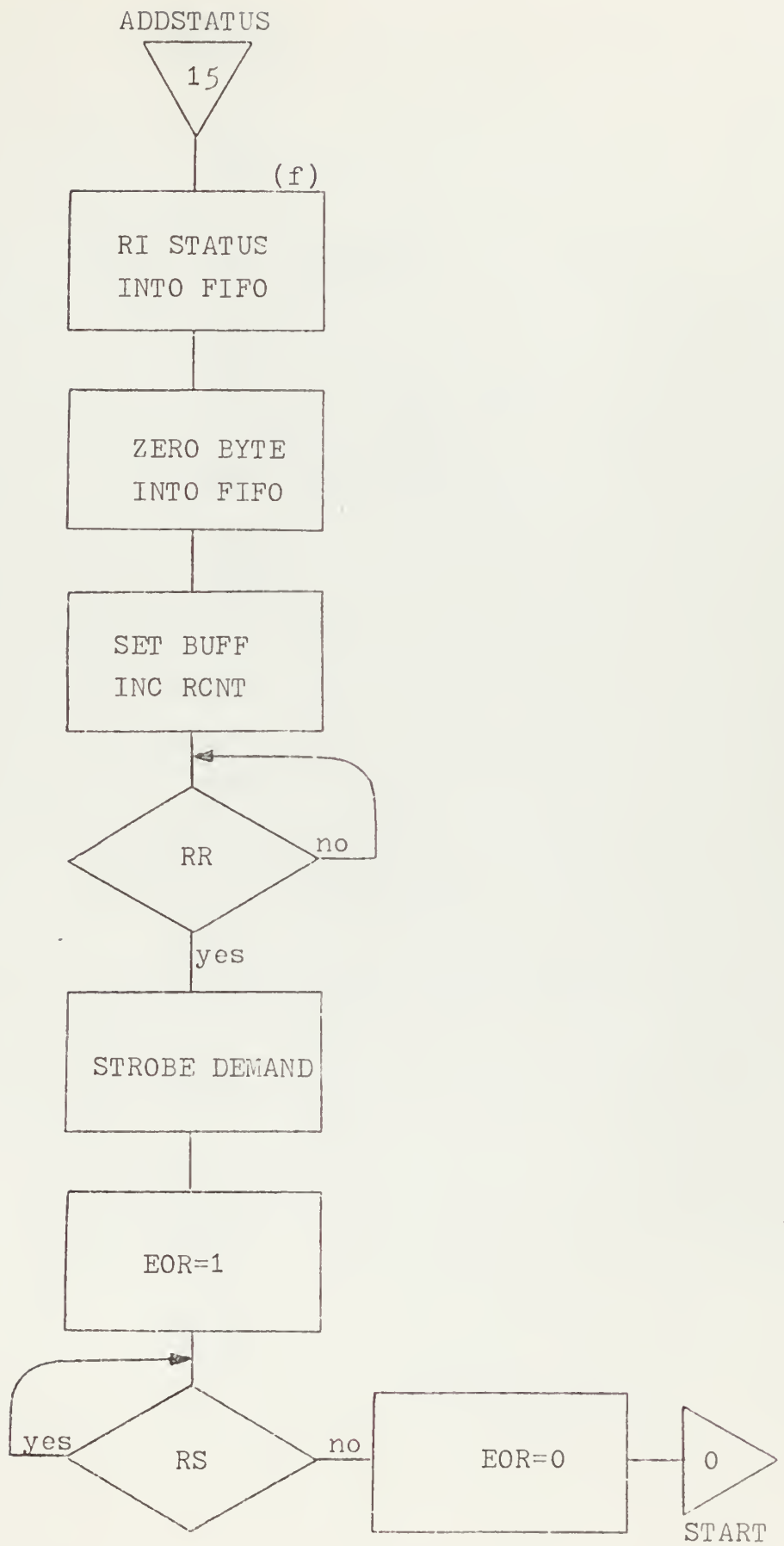


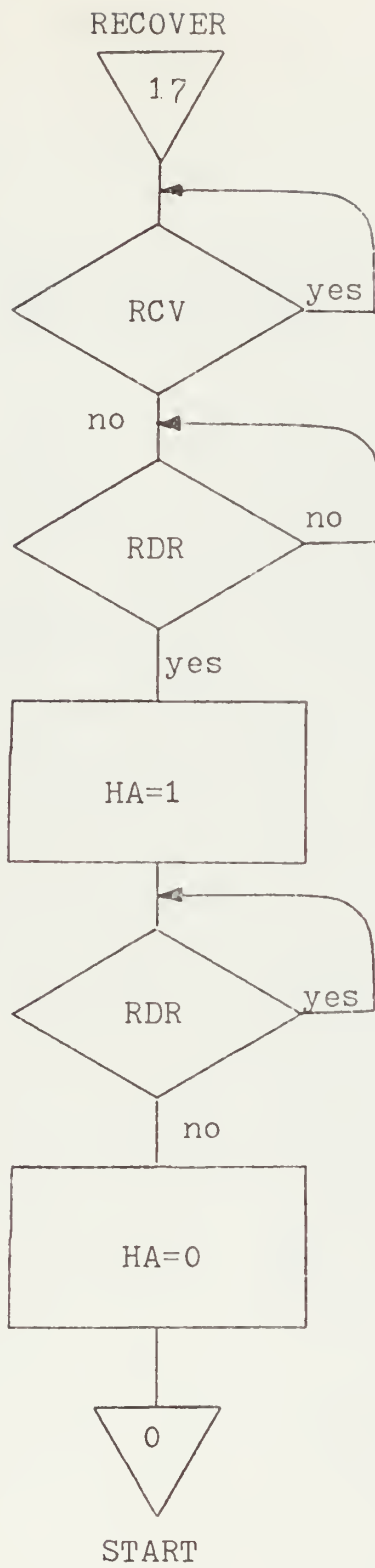


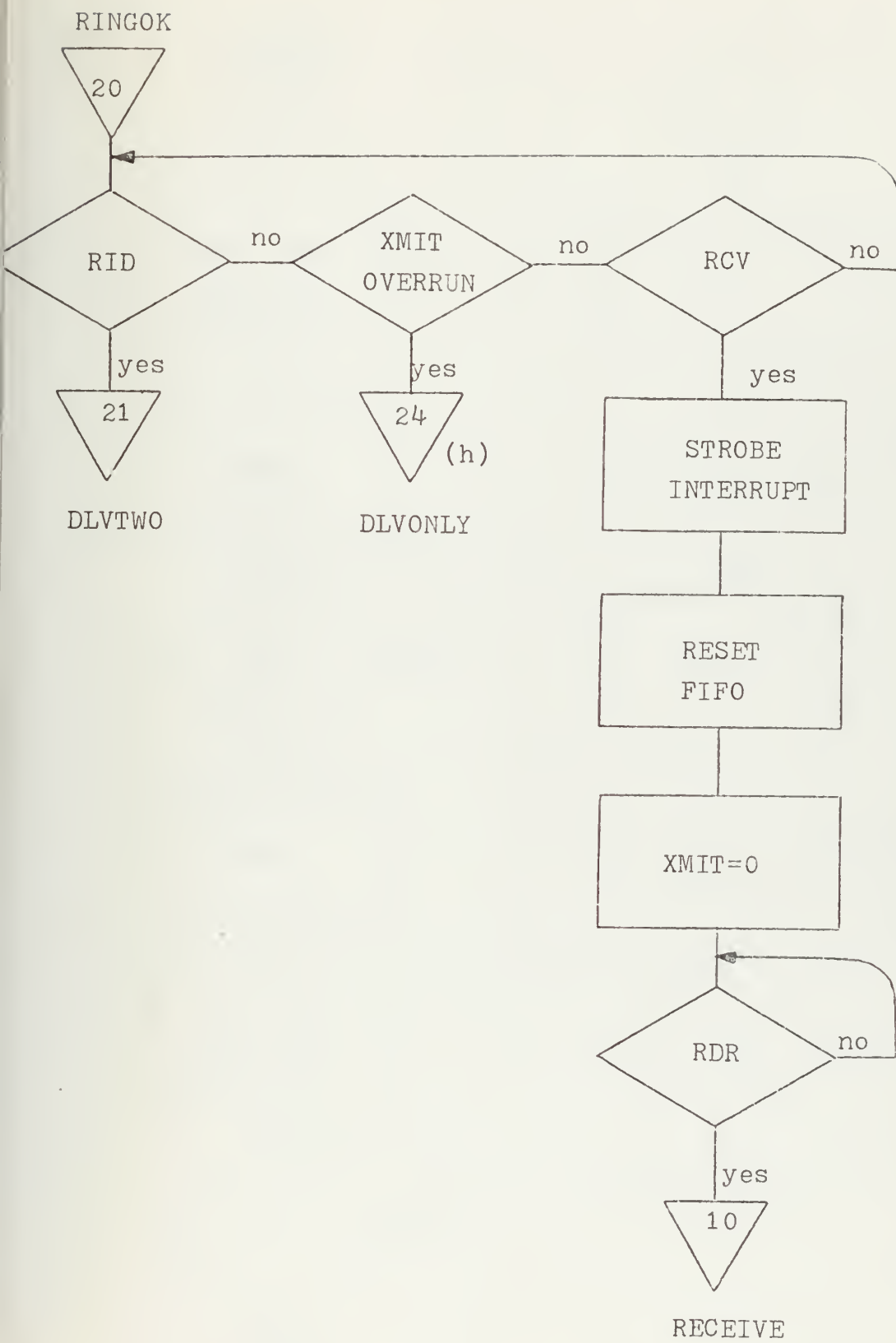


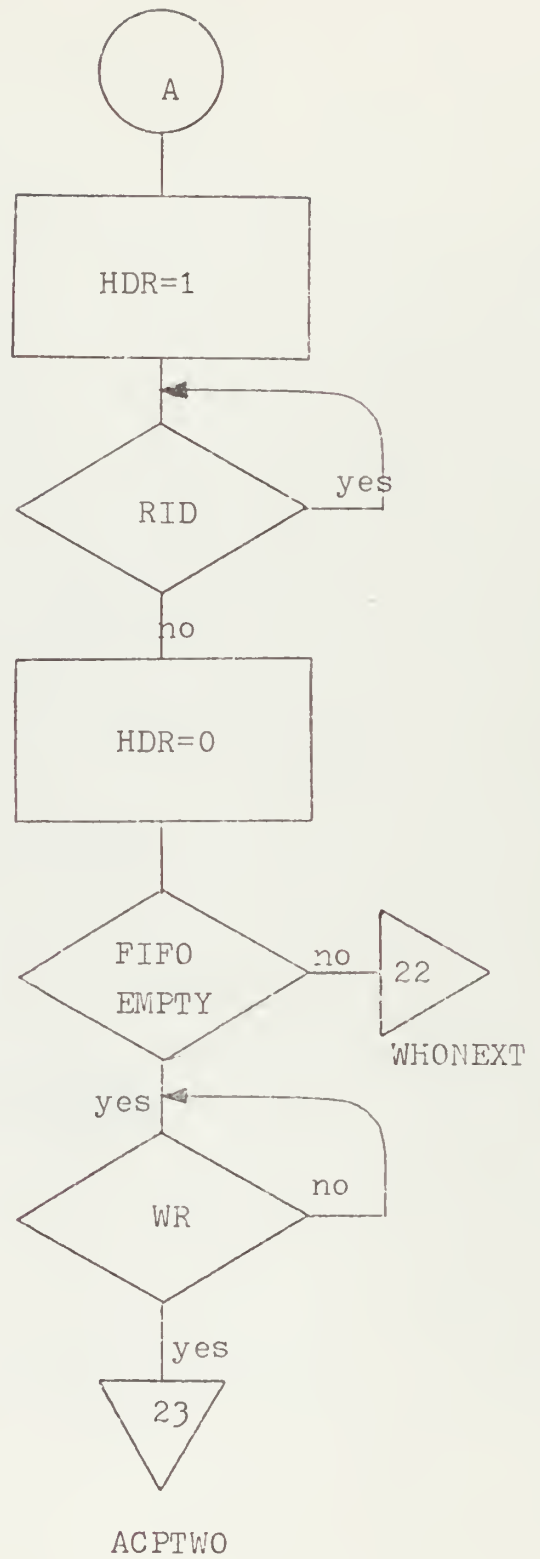
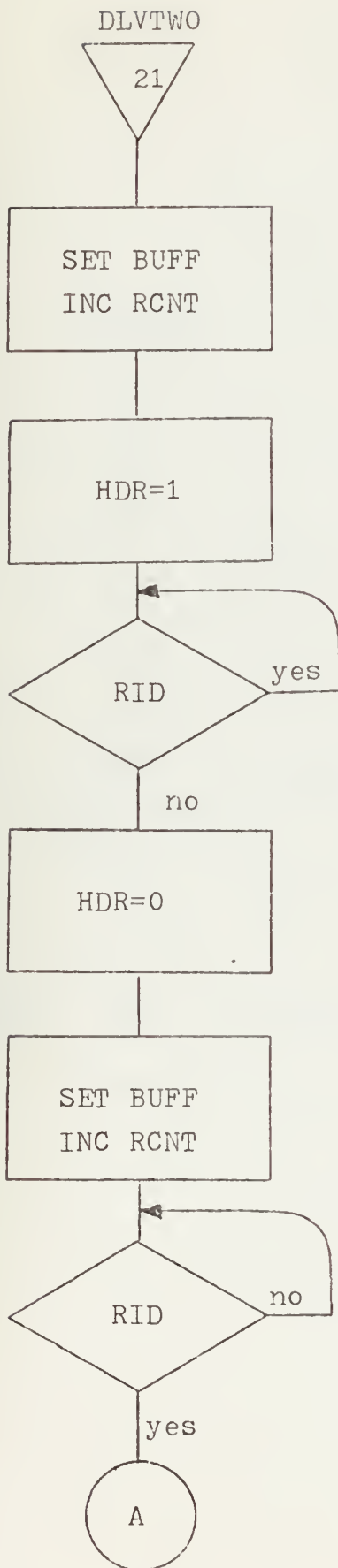


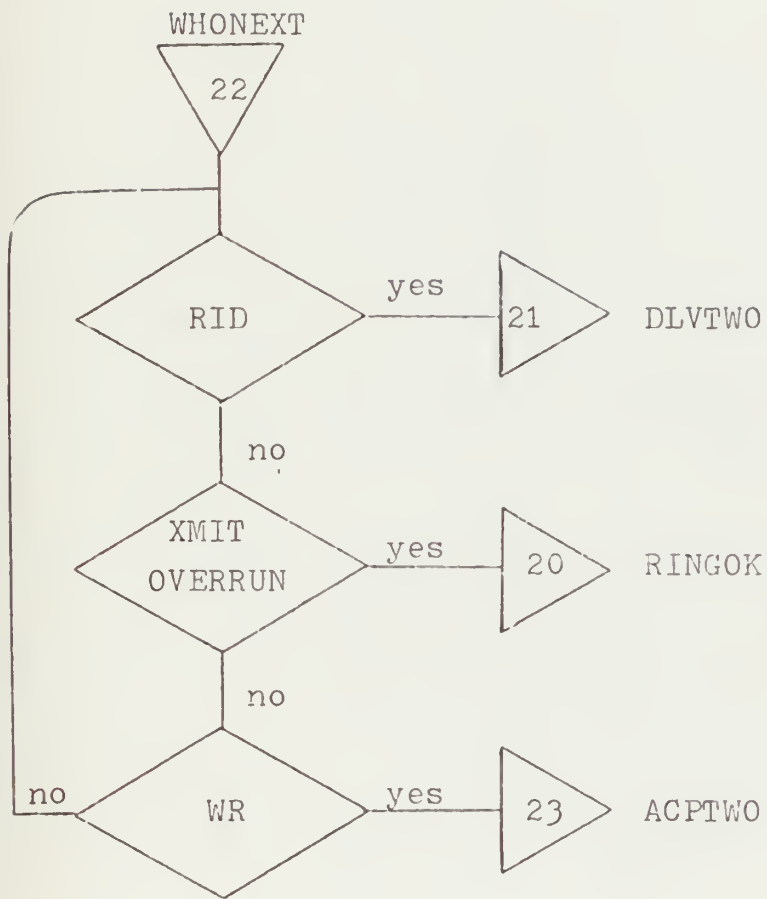


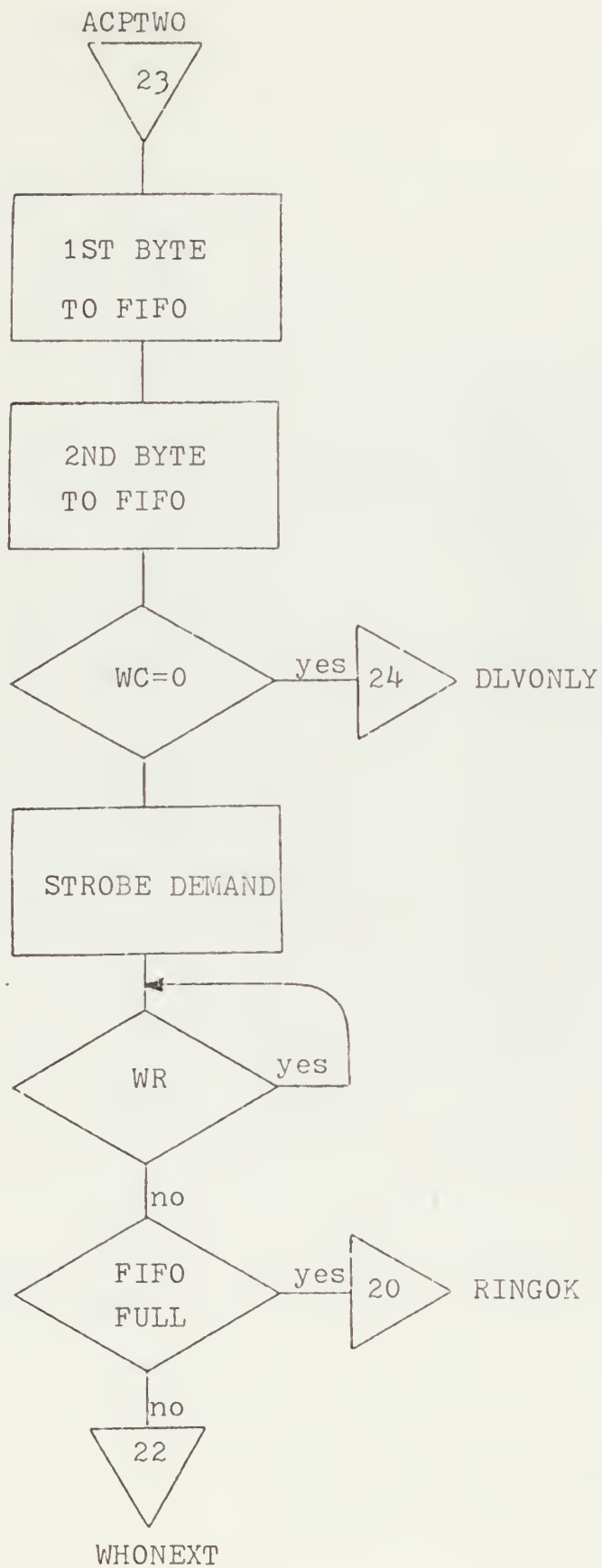


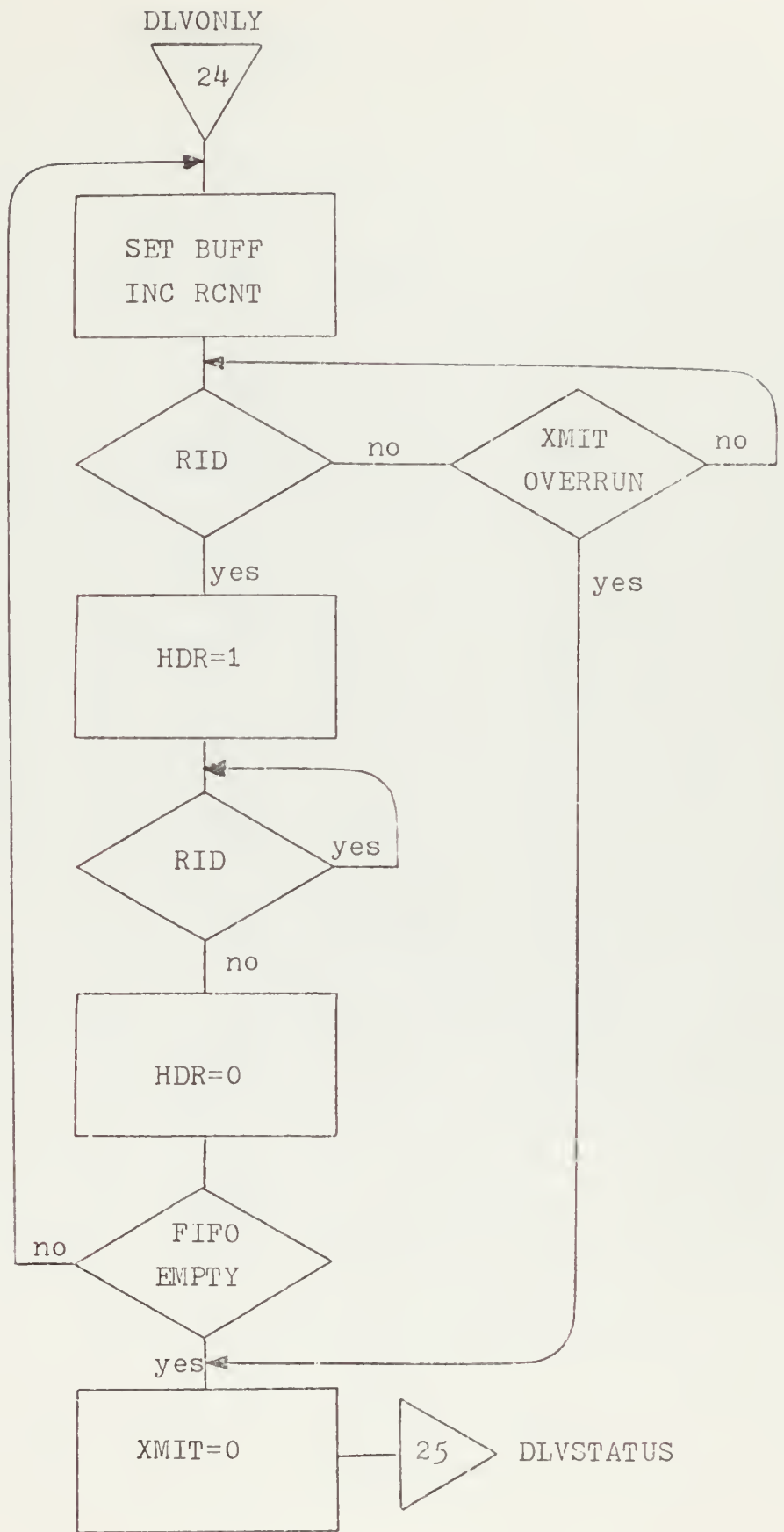


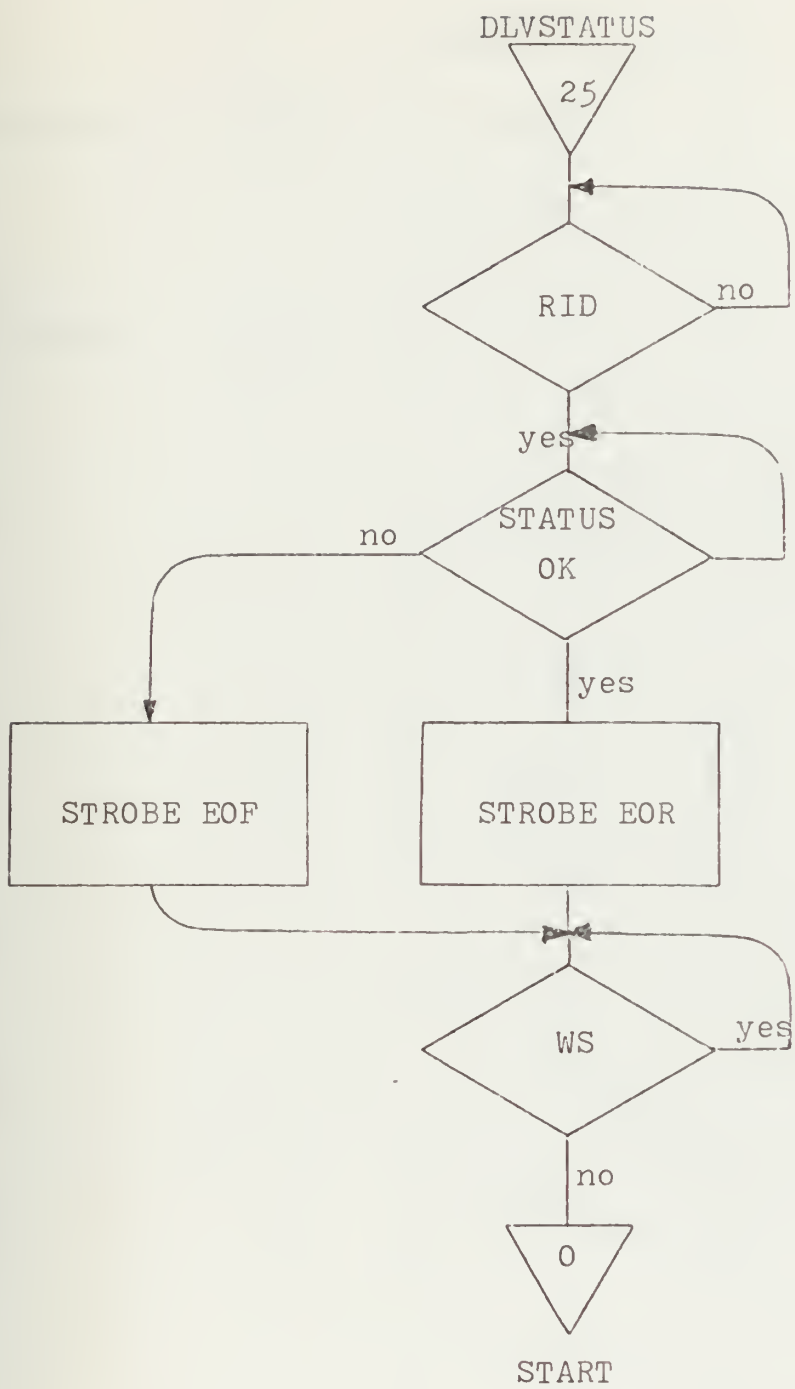


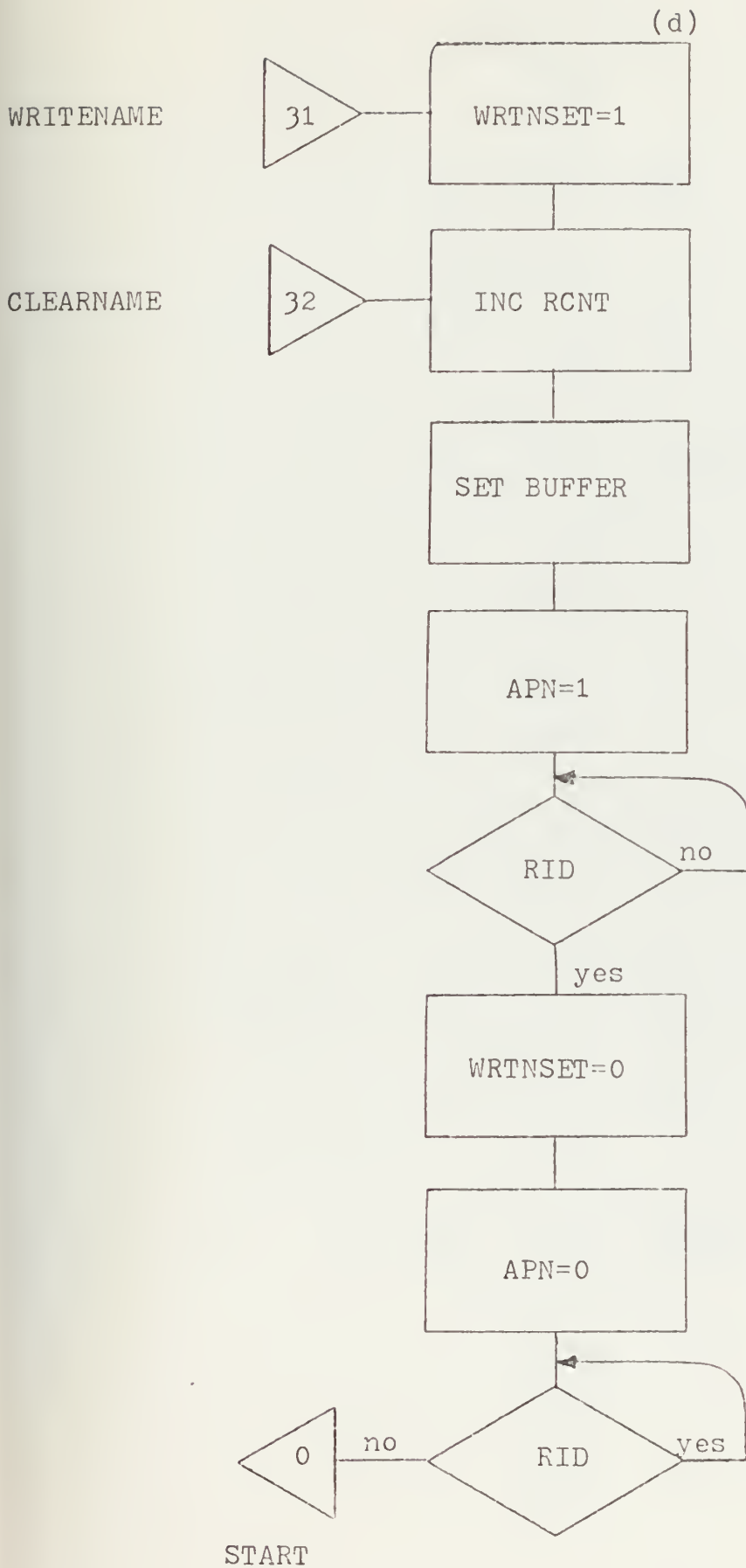




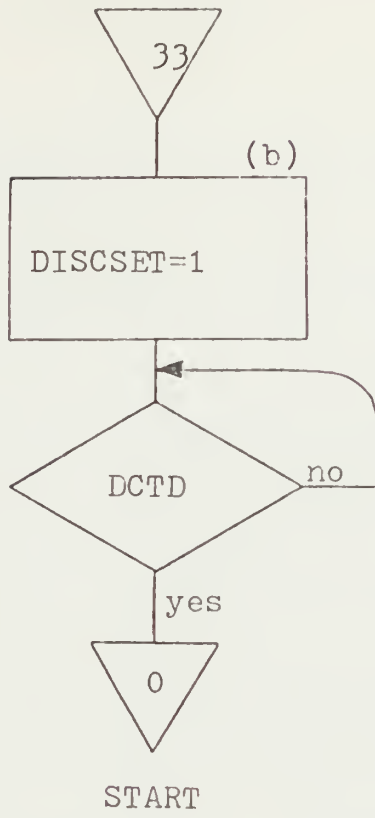




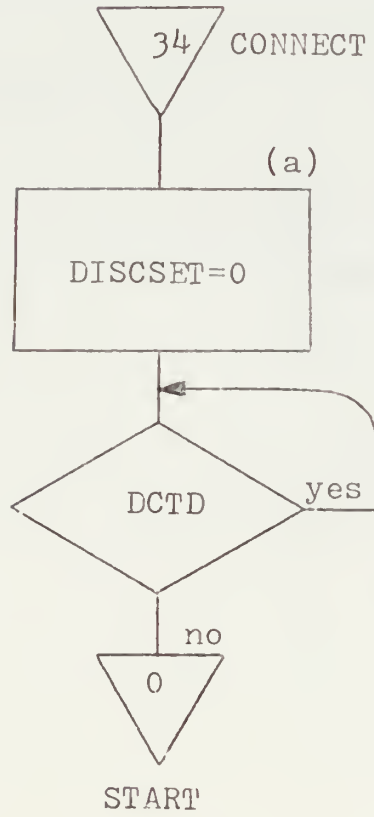


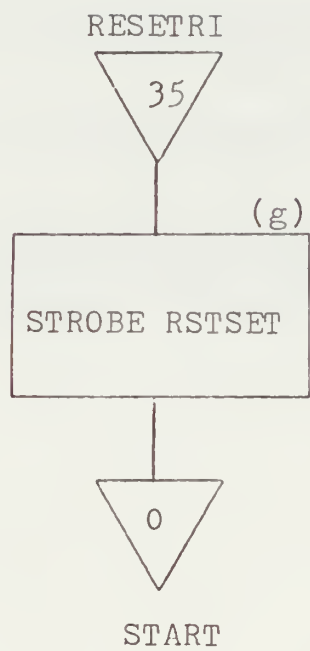
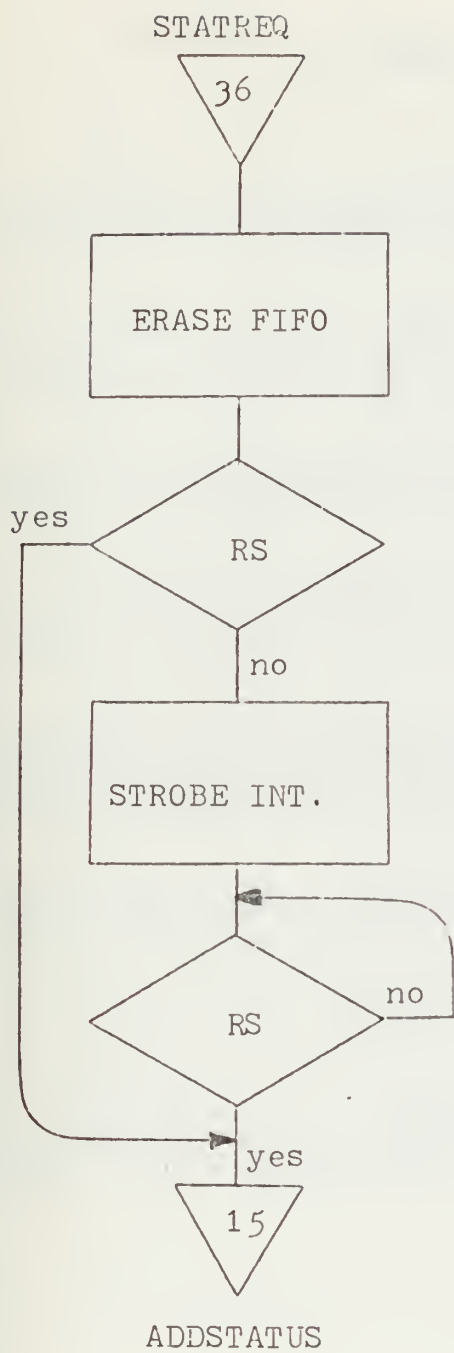


DISCONNECT



CONNECT





BIBLIOGRAPHY

1. Brubaker, R. H. , A GENERAL PURPOSE MICROCONTROLLER, paper prepared at Computer Science Group (internal document), Naval Postgraduate School, Monterey, California, 1974.
2. Brubaker, R. H. , PROPOSED RING REPEATER DESIGN, paper prepared at Computer Science Group (internal document), Naval Postgraduate School, Monterey, California, 1974.
3. Harris, M. S. , A PROTOTYPE RING INTERFACE FOR THE NPS DATA COMMUNICATION RING, Master's Thesis, Naval Postgraduate School, Monterey, California, 1974.
4. Hirt, K. A. , A PROTOTYPE RING-STRUCTURED COMPUTER NETWORK USING MICROCOMPUTERS, Master's Thesis, Naval Postgraduate School, Monterey, California, 1973.
5. Johnson, R. P. and Kirkham, J. H. , THE SYSTEM 360 SOFTWARE INTERFACE FOR THE NAVAL POSTGRADUATE SCHOOL COMPUTER RING SYSTEM, Master's Thesis, Naval Postgraduate School, Monterey, California, 1974.
6. Kildall, G. A. , A SYMBOLIC MICROCONTROLLER ASSEMBLY LANGUAGE, paper prepared at Computer Science Group (internal document), Naval Postgraduate School, Monterey, California, 1974.
7. Loomis, D. C. , RING COMMUNICATIONS PROTOCOLS, U. C. Irvine Distributed Computer Project, Memo 46-A, 1972.
8. Wortmann, E. O. , DESIGN AND IMPLEMENTATION OF A RING INTERFACE/HOST ADAPTOR FOR AN IBM SYSTEM 360, Master's Thesis, Naval Postgraduate School, Monterey, California, 1974.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Library (Code 0212) Naval Postgraduate School Monterey, California 93940	2
2. Chairman, Computer Science Group (Code 72) Naval Postgraduate School Monterey, California 93940	1
3. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
4. LT(jg) Gary M. Raetz, USN (Code 72 Rr) Computer Science Group Naval Postgraduate Group Monterey, California 93940	1
5. Professor V. M. Powers (Code 72 Pw) Computer Science Group Naval Postgraduate School Monterey, California 93940	1
6. LCDR J. S. Meserve, USN 5624 Jockey Way Bonita, California 92002	1
7. LT. R. E. Stewart, USN 3003 Corral Court San Jose, California 95121	1

Thesis

M548

Meserve

c.1

Implementation and
testing of the NPS Data
Communication Ring.

161003

3

22 JUL 76

10 MAY 77

01 MAY 77

07 OCT 77

11 AUG 81

28 NOV 83

23800

23748

24421

24421

24-15

26802

27906

161003

Thesis

M548

Meserve

c.1

Implementation and
testing of the NPS Data
Communication Ring.

thesM548

Implementation and testing of the NPS Da



3 2768 001 88277 2

DUDLEY KNOX LIBRARY